2

TEXAS A & M UNIVERSITY

Texarkana Division

Texarkana, Texas 75501

Report DARCOM-ITC-02-08-76-008

REGRESSION WITH DIFFERENTIAL EQUATION MODELS

Craig D. Hunter
Maintenance Effectiveness Engineering Graduate Center
DARCOM Intern Training Center
Red River Army Depot
Texarkana, Texas 75501

April 1976

Final Report

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

Prepared for

MAINTENANCE EFFECTIVENESS ENGINEERING GRADUATE
PROGRAM AND TEXAS A&M UNIVERSITY GRADUATE CENTER
DARCOM Intern Training Center
Red River Army Depot
Texarkana, Texas 75501

FOREWORD

The research discussed in this report was accomplished as part of the Maintenance Effectiveness Engineering Graduate Program conducted jointly by the DARCOM Intern Training Center and Texas A&M University. As such, the ideas, concepts and results herein presented are those of the author and do not necessarily reflect approval or acceptance by the Department of the Army.

This report has been reviewed and is approved for release. For further information on this project contact Dr. Ronald C. Higgins, Intern Training Center, Red River Army Depot, Texarkana, Texas 75501.

APPROVED:

Ronald C. Higgins

Dr. Ronald C. Higgins, Chief
Maintenance Effectiveness Engineering

For the Commander

James L. Arnett

James L. Arnett, Director
Intern Training Center

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

14

1. ~~REPORT NUMBER~~

DARCOM-ITC-02-08-76-008

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE *(and Subtitle)*

6

REGRESSION WITH DIFFERENTIAL EQUATION MODELS

5. ~~TYPE OF REPORT & PERIOD~~ COVERED

9

Final rept.

6. ~~PERFORMING ORG. REPORT~~ NUMBER

7. AUTHOR(s)

10

Craig D. Hunter

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS

Department of Maintenance Effectiveness
DARCOM Intern Training Center
Red River Army Depot, Texarkana, Texas 75501

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS

Department of Maintenance Effectiveness and
Texas A&M University Graduate Center
DARCOM Intern Training Center

12. ~~REPORT DATE~~

11

Apr[redacted] [redacted]76

12

99p.

13. ~~NUMBER OF PAGES~~

91

14. MONITORING AGENCY NAME & ADDRESS *(If different from Controlling Office)*

15. SECURITY CLASS. *(of this report)*

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

Research performed by Craig D. Hunter under the supervision of Dr. S. Bart Childs, Adjunct Professor, Texas A&M University.

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Differential equation regression.

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

Regression analysis normally implies the use of algebraic equations to describe a system; however, some cases would better be modeled by differential equations. This is accomplished by assuming a differential equation model for a given set of data and estimating the values of the unknown parameters within the model. These values are then systematically perturbed to generate particular solutions which are superimposed to yield a better estimate of the unknowns. This process is repeated until a specified accuracy is met.

ACKNOWLEDGMENTS

I would like to thank Dr. Bart Childs for his endless help and guidance without which this paper could not have been completed. The support of Dr. J.W. Foster and Dr. E.A. Timsans was also appreciated as well as the help in structure and style by Dr. D.A. Bremmer. To those who helped in debugging portions of the computer programming, thank-you. The many typings, revisions, format, and grammatical corrections could not have been done without the help of Ann Silvernail and Karon Potter.

During the course of this study, my employer was the U.S. Army as a career intern in the DARCOM Maintenance Effectiveness Engineering Graduate Program. I am grateful to the U.S. Army for the opportunity to participate in this program.

ABSTRACT

Research Performed by Craig D. Hunter

Under the Supervision of Dr. S. Bart Childs

Regression analysis normally implies the use of algebraic equations to describe a system; however, some cases would better be modeled by differential equations. This is accomplished by assuming a differential equation model for a given set of data and estimating the values of the unknown parameters within the model. These values are then systematically perturbed to generate particular solutions which are superimposed to yield a better estimate of the unknowns. This process is repeated until a specified accuracy is met.

Through an analysis of variance, the statistical characteristics of linear regression can be generated for most n^{th} order differential equations. This provides a basis for evaluating the 'acceptance or rejection' of the regression.

The characteristics generated consist of an ANOVA table (uncorrected), general F test on the regression, the R^2 value, covariance matrix of the superposition constants, an estimate of the variance about the regression, an estimate of the variance of the parameters, and the confidence intervals on these estimates.

Through an analysis of variance, the statistical characteristics of linear regression can be generated for most n^{th} order differential equations. This provides a basis for evaluating the 'acceptance or rejection' of the regression.

The characteristics generated consist of an ANOVA table (uncorrected), general F test on the regression, the R^2 value, covariance matrix of the superposition constants, an estimate of the variance about the regression, an estimate of the variance of the parameters, and the confidence intervals on these estimates.

LIST OF CONTENTS

Chapter		Page
I	INTRODUCTION	1
II	DIFFERENTIAL REGRESSION	3
III	THE OVER-DETERMINED SYSTEM	8
IV	ANALYSIS OF VARIANCE	10
	Table 4.1 ANOVA Table	11
V	RESULTS-COASTING DATA	14
	Figure 5.1 Input of Coasting Data	16
	Figure 5.2 PSTAT Output	17
VI	CONCLUSIONS ON STATISTICAL RESULTS	19
	REFERENCES	20
	APPENDICES	22
	A. Nomenclature	23
	B. General Linear Regression Theory (Draper and Smith, 1966)	25
	Table B.1 ANOVA Table	26
	C. Applications to Algebraic Regression	29
	D. The Control Program with Output of the Non-Linear Mass-Spring-Damper System	31

CHAPTER I

INTRODUCTION

Regression analysis usually implies determining a set of independent variables and corresponding coefficients for algebraic equations such that these equations best portray the data or system given. Examples are presented in which algebraic models are inadequate and can be appropriately modeled by differential equations.

This report is an investigation into the validity and accuracy of a differential equation regression package, both linear and non-linear (Childs, 1970). It is not the intent to deal with extensions of the established theory behind such an approach (Bellman and Kalaba, 1965), (Roberts and Shipman, 1972). Nomenclature and background material are given in Chapter II.

A control program was written to provide a means for evaluating QUASII (Childs, 1970). The control program yields solutions of a specified accuracy (within the computational restrictions of the computer) providing a means for comparison. QUASII would be used instead of a program yielding "exact" solutions because of ease of use.

The control program is based on power series evaluation and integration of differential equations; whereas, QUASII utilizes a 4th order Runge-Kutta integrator. The Runge-Kutta integrator allows simpler programming of non-linear differential equations.

Two solution areas are of interest:

Case I: The number of boundary conditions (m) equals the order of the differential equation (n), therefore all boundary conditions are to be

met exactly (exact system).

Case II: The number of boundary conditions (m) exceeds the order of the differential equation (n), therefore some boundary conditions are to be met exactly and others to be met in a least squares sense (over-determined system).

Both QUASII and the control program utilize a least squares fit of the non-exact boundary conditions in Case II. Further study in this area is being done by Walker (Ref. 21). These results will be available in April 1976 and entails meeting these boundary conditions in a least squares and other criteria.

Case I is of theoretical interest. This report contains results for Case II studies.

CHAPTER II

DIFFERENTIAL REGRESSION

The following example of a linear case from Childs et. al. 1971, (Ref. 6), is for clarification of terms and background. For information on non-linear cases, refer to Reference 5 and 17.

Given: A simple mass-spring-damper system governed by the equation;

$$\ddot{x} + \mu\dot{x} + \xi x = \sin(wt) \tag{2-1}$$

where $(\cdot)$ denotes the total derivative of that term with respect to t, the independent variable.

Denoting the following state variables;

$$y_1 = x \qquad y_2 = \dot{x} \tag{2-2}$$

equation 2-1 can be rewritten as two non-trivial first order differential equations:

$$\begin{aligned} \dot{y}_1 &= y_2 \\ \dot{y}_2 &= -\xi y_1 - \mu y_2 + \sin(wt) \end{aligned} \tag{2-3}$$

The state variables y_1 and y_2 are functions of t, $y_i(t)$ (t has been omitted for simplicity).

If μ and ξ are known, equations 2-1 and 2-3 are linear differential equations. Assuming that μ and ξ are unknown, which is not an unreasonable assumption, two more state variables must be added;

$$y_3 = \mu \qquad y_4 = \xi$$

of which the following four equations result;

$$\dot{y}_3 = 0 \tag{2-4}$$

$$\dot{y}_4 = 0 \tag{2-5}$$

$$\dot{y}_1 = y_2 \tag{2-6}$$

$$\dot{y}_2 = -y_1y_4 - y_2y_3 + \sin(wt) \tag{2-7}$$

the frequency w could have been assumed unknown, introducing a fifth state variable, $y_5 = w$. Except for equation 2-7, the four differential equations are linear eith 2-4 and 2-5 being trivial. Since equation 2-7 is non-linear, the system is non-linear.

At least two Newton type or Taylor series methods used in solving such systems are perturbation and quasilinearization methods.

Perturbation methods are used in the control program and QUASII for ease of programming. For information regarding this theory, see Reference 5, Doiron (1970), or any of Childs' writings on non-linear differential equation solutions.

The right hand side (RHS) of equation 2-3 is unknown (except for μ and ξ), therefore, avenues are opened for the implementation of boundary conditions or constraints which must be met. In a mass-spring-damper system governed by equation 2-1, the use of an accelerometer to measure the acceleration of the mass at certain points in time (t_i) is within experimental procedure. An accumulation of constraints (boundary values) could be obtained in the following form;

$$\ddot{x}(t_i) = b_i \qquad i = 1 \ldots m \tag{2-8}$$

where b_i is the measured acceleration at time t_i and m is the number of boundary values. The LHS of equation 2-3, y_2 ($y_2 = \ddot{x}$), can now be replaced by b_i;

$$b_i = -\xi y_1(t_i) - \mu y_2(t_i) + \sin(wt_i) \tag{2-9}$$

in more general terms;

$$q_i(y(t_i)) = b_i \qquad i = 1...m \qquad (2\text{-}10)$$

where q_i is a boundary condition operator. Equation 2-10 defines m boundary conditions resulting in m linear (or as in equation 2-7, non-linear) equations. From Chapter I, m must be equal to or greater than n, the order of the differential equation (number of state variables). These m differential equations will either be met exactly or at least in a least square (best fit) manner.

A brief summary of a "shooting" technique of solving linear differential equations follows. Given;

$$\dot{y} = Ly + f \qquad (2\text{-}11)$$

where L is a linear operator and f is a forcing function.

A set of homogeneous solutions is usually superimposed upon a particular solution. The number of homogeneous solutions is equal to n, the order of the differential equation (see equation 2-1, n=2). Assuming that no initial conditions are known, n = r (let r equal the number of unknown initial conditions, where $r \leq n$). When only r initial conditions are unknown, r homogeneous equations are required. The result of the superposition of a particular solution (2-12) and a set of homogeneous equations (2-13) is:

$$\dot{p} = Lp + f \qquad (2\text{-}12)$$

$$\dot{H} = LH \qquad (2\text{-}13)$$

$$y = p + H\beta = p + \sum_{k=1}^{r} h^{(k)}\,\beta k \qquad (2\text{-}14)$$

It is an established theory that the sum of a particular solution and a set of homogeneous equations yields another particular solution. An alternative is to superimpose only particular solutions. This scheme

is used in QUASII and the control program;

$$y = Pa = \sum_{k=0}^{r} p^{(k)} a_k \tag{2-15}$$

where

$$\dot{p}^{(k)} = Lp^{(k)} + f \tag{2-16}$$

There will be r+1 particular solutions, where r is the number of unknown initial values.

A restriction on the superposition constants is:

$$\sum_{k=0}^{r} a_k \equiv 1 \tag{2-17}$$

This is a result of equation 2-11, where the differential equation is given by:

$$\dot{y} = Ly + f$$

If particular solutions of the form of equation 2-16 are superimposed without the constraint 2-17, the resulting differential equation could be

$$\dot{y} = Ly + Cf \tag{2-18}$$

where

$$c = \sum_{k=0}^{r} a_k \neq 1 \tag{2-19}$$

The original differential equation has not been satisfied and necessitates the restriction on the superposition constants to yield

$$\dot{y} = Ly + f \sum_{k=0}^{r} a_k \tag{2-20}$$

The computational strategy requires choosing initial arbitrary values for the r unknown state variables. These values are used in defining the unperturbed particular solution (denoted by $p^{(o)}$). Each of the unknown variables (r of them) will in turn be individually perturbed by a

predetermined (aribtrary) constant to yield an appropriately perturbed particular solution. These particular solutions, obtained by integration, yield values corresponding to the boundary values at the appropriate value of the independent variable.

The result is a matrix equation of the form

$$Sa = d \tag{2-21}$$

where S is a function of the integrated solutions, a is the vector of superposition constants, and d is a vector of boundary values. The resulting m+1 equations (m due to the boundary conditions and one due to the restriction on the superposition constants) contain r+1 unknown a_k's ($r+1 \leq m+1$), which can be determined as discussed in Chapter III.

The resulting y solutions (state variables) are the initial arbitrary estimates plus the addition of the corresponding superposition constant times the perturbation constant.

Evaluation of the a's was stated to be a rather simple process. This entails a Gauss-Jordan reduction utilizing a least squares fit of the over-determined system. A more in-depth discussion of the theory is presented in Chapter III and Reference 4.

CHAPTER III

THE OVER-DETERMINED SYSTEM

When the number of boundary conditions (m) equals the order (n) of the differential equation, the boundary conditions can theoretically all be met exactly, however, in the case of the over-determined system, this is impossible. Thus, several boundary values will have to be met by some other means.

The problem of Chapter II was narrowed to the final equation:

$$Sa = d \tag{3-1}$$

This equation and the role of a least squares fit are the topics of this chapter.

Examining the composition of the S matrix, two subdivisions can be shown;

$$\begin{array}{cc} \begin{bmatrix} S_A \\ \hline S_B \end{bmatrix} & \begin{array}{c} k+1 \\ \hline m-k \end{array} \\ r+1 & \end{array}$$

where k is the number of boundary conditions to be met exactly. The S_A matrix contains the exact conditions and S_B the least squares boundary conditions.

A further reduction or partitioning can be;

$$\left[\begin{array}{c|c} S_1 & S_2 \\ \hline S_3 & S_4 \end{array}\right] \begin{bmatrix} a \end{bmatrix} = \begin{bmatrix} d \end{bmatrix} \tag{3-2}$$

where

S_1 is order (k+1) x (k+1)

S_2 is order (k+1) x (r-k)

S_3 is order (m-k) x (k+1)

S_4 is order (m-k) x (r-k)

By a Gauss-Jordan reduction, the S_1 matrix can be transformed into an identity matrix resulting in S_3 being a null matrix. The results are;

$$\left[\begin{array}{c|c} I & S_2' \\ \hline 0 & S_4' \end{array}\right]\left[\begin{array}{c} a_e \\ \hline a_l \end{array}\right]=\left[\begin{array}{c} d_e' \\ \hline d_l' \end{array}\right] \tag{3-3}$$

where the subscripts l and e refer to least square and exact boundary conditions respectfully.

The null matrix (S_3) allows for an equation (3-4) containing only least squares conditons.

$$S_4'a_l = d_l' \tag{3-4}$$

Equation 3-4 yields m-k equations each with an unknown.

Premultiplying by the transpose of S_4' gives the normal equation (refer to Appendix B);

$$(S_4')^T S_4'a_l = (S_4')^T d_l' \tag{3-5}$$

resulting in the solution for a_l:

$$a_l = \left[(S_4')^T S'_4\right]^{-1}(S_4')^T d_l' \tag{3-6}$$

Having solved for a_l in equation 3-6, a_e (from exact conditions) can be solved by referring to equation 3-3;

$$Ia_e + S_2'a_l = d_e' \tag{3-7}$$

or

$$a_e = d_e' - S_2'a_l \tag{3-8}$$

Equation 3-4 and 3-5 are important and will be used extensively in Chapter IV.

CHAPTER IV

ANALYSIS OF VARIANCE

The boundary conditions are:

$$q(y(t_i)) = b_i \qquad i = 1...m \tag{4-1}$$

This defines an operator q operating on the $y(t_i)$'s to yield the desired boundary values. The statistical basis for 'accepting or rejecting' the regression is the nearness of the equality 4-1.

In Appendix B, the equation;

$$z = Wd + \varepsilon \tag{4-2}$$

is analyzed through an analysis of variance.

From equations 4-1 and 4-2, the following results can be obtained;

$$b = \hat{q} + \varepsilon \tag{4-3}$$

where

$$\hat{q} = q(\hat{y})$$

$$b \approx z$$

$$\varepsilon = \varepsilon$$

$$\hat{q} \approx Wd$$

Note that $q(y(t_i))$ may not be linear as required for an analysis of variance as described in Appendix B, but the solution process is iterative using linearized equations. Because the steps are small, the assumption of $\hat{q}$ being linear is a good approximation.

An analysis of variance can be accomplished as shown in Table 4.1.

TABLE 4.1

ANOVA TABLE

Source	Sum of Squares	Degrees of Freedom	Mean Square
Due to regression	$\hat{q}^T b$	r	SS/r
About the regression (residual)	$b^T b - \hat{q}^T b$	m-k-r	s^2
Total (uncorrected)	$b^T b$	m-k	

The following are calculated directly from Table 4.1:

$$R^2 = \frac{\text{SSregression} - (m-k)(\text{mean of } b_i\text{'s})^2}{\text{SStotal} - (m-k)(\text{mean of } b_i\text{'s})^2} \quad (4\text{-}4)$$

$$F_{cal} = \frac{\text{MS regression}}{\text{MS residual}} \quad (4\text{-}5)$$

$$s^2 = \text{MS residual} = \text{estimated variance of system} \quad (4\text{-}6)$$

The F_{cal} value will be compared to a Fisher's F with:

Probability of $1-\alpha$ (α is the producer's risk)

Numerator degrees of freedom r

Denominator degrees of freedom (m-k-r)

If F_{cal} exceeds Table F, the regression is 'accepted'. (Note: The producer's risk is defined as the probability of rejecting the null hypothesis (H_0) (see Appendix B) when in fact the null hypothesis is true.

Confidence limits for the estimated least square boundary values can be determined once the individual variances are known. Referring to Appendix B, the variance of the estimated observation is given by equation B-10:

$$\text{est. var}(\hat{z}_i) = W_i(W^TW)^{-1}W_i^Ts^2 \tag{4-7}$$

Equation 4-3 shows that $\hat{q}$ is synonymous with Wd of Appendix B, but $\hat{q}$ cannot be subdivided into a matrix of independent variables and a vector of parameters, therefore another approach must be used.

Using the system of equation 3-5, a relation between d_1 (least squares boundary values of b) and d_1' (transformation ot d_1) can be determined.

The variances of the individual least squares boundary values;

$$\left[S_3 \;\vdots\; S_4\right] \left[a\right] = \left[d_1\right] \tag{4-8}$$

is equal to the variances of the transformed system:

$$\left[0 \;\vdots\; S_4'\right] \left[a\right] = \left[d_1'\right] \tag{4-9}$$

i.e.

$$\text{var}(d_1) = \text{var}(d_1') \tag{4-10}$$

The variances of the estimated least squares boundary values are given by;

$$\text{est. var}(\hat{b}_i) = S_{4_i}' \left[(S_4')^T S_4'\right]^{-1} (S_{4_i}')^T s'^2 \quad i=1\ldots m \tag{4-11}$$

where S_{4_i}' is the row vector of S_4' corresponding to the i^{th} least squar[illegible] boundary condition.

The resulting confidence limits are;

$$\hat{b}_i \pm t(\nu, 1-\tfrac{\alpha}{2}) \sqrt{\left[\text{est. var}(\hat{b}_i)\right]} \tag{4-12}$$

where

$$\nu = \text{m-k-r} \tag{4-13}$$

$$i = k+1 \ldots m$$

A main objective of the regression package is to obtain the solutions

for the y_i's given in Chapter II. Table 4.1 does not always provide a means for investigating this because it involves the boundary condition functions. By doing a partial analysis of variance on equation 4-10, the variance of the a_1's can be determined;

$$\text{est. } var(a_1) = ((S_4')^T S_4')^{-1} s'^2 \qquad (4\text{-}14)$$

where

$$s'^2 = \frac{(d_1')^T(d_1') - a_1^T(S_4')^T d_1'}{m-k-r} \qquad (4\text{-}15)$$

Referring to Chapter II, the y_i solutions are equal to the previous value (initial guess) plus the perturbation times a superposition constant;

$$\hat{y} = \hat{y}_{previous} + (a_1 * \text{perturbation}) \qquad (4\text{-}16)$$

therefore

$$var(\hat{y}) = var(a_1) * (\text{perturbation})^2 \qquad (4\text{-}17)$$

This results in a confidence interval for these solutions as follows:

$$\hat{y}_i \pm t(\nu, 1-\tfrac{\alpha}{2}) \sqrt{\left[\text{est. } var(\hat{y}_i)\right]} \qquad i = 1 \ldots r \qquad (4\text{-}18)$$

CHAPTER V

RESULTS-COASTING DATA

The statistical study outlined in Chapter IV was accomplished by a subroutine package PSTAT (Appendix D contains PSTAT and supportive subroutines).

Since experimental data was not available on the system of equation 2-1, a problem of determining the aerodynamic drag coefficient and rolling friction coefficient of an automobile was used (Ref. 21). The nonlinear equation governing this system is;

$$\ddot{x} + \frac{A_f \rho}{2M} C_d \dot{x}^2 + \mu_f g = 0 \qquad (5\text{-}1)$$

where

ρ = air density (slugs/ft^3)

A_f= frontal area of vehicle (ft^2)

M = mass of vehicle (slugs)

g = acceleration of gravity (ft/sec^2)

C_d= coefficient of drag

μ_f= rolling friction coefficient

$\dot{x}$ = velocity (ft/sec)

$\ddot{x}$ = acceleration (ft/sec^2)

This problem demonstrates the advantages of modeling by differential equations with a regression package like QUASII. Attempts to determine C_d and μ_f by other than numerical methods (such as wind tunnel testing, treadmills, etc.) would be more laborious, less accurate, and more expensive.

Coasting data (value of $\dot{x}$) for a Sunbeam Alpine was obtained from

Road and Track Road Test Annual for 1966. Figure 5.1 is a listing of input for such a system. Figure 5.2 is an output of the statistical analysis from PSTAT. Figure 5.1 shows only one non-trivial differential equation. Although equation 5-1 is a 2nd order differential equation, it does not contain an x term, allowing a single first order representation.

Except for three parameters, the PSTAT output (Figure 5.2) has been previously defined. The three additional parameters are:

1) Mean of the observations is the sum of the least squares observations (boundary values) divided by the total number of these observations:

$$\bar{b} = \frac{\sum_{i=1}^{m-k} b_i}{m-k} \tag{5.2}$$

2) Coefficient of variation (cv) is a measure of the dispersion of the data (Hald, 1952) and is expressed as:

$$c = \frac{\text{MS residual}}{\text{mean of observations}} = s/\bar{b} \tag{5-3}$$

The importance of this simple parameter is questionable.

3) P(F(ALPHA).GT.FCAL) is the probability that a random variable that is F distributed, with the same degrees of freedom as F(ALPHA), will be greater than FCAL, thus the minimum producer's risk that can be assumed.

The PSTAT output (Figure 5.2) is an ANOVA table of the uncorrected sum of squares plus the corresponding degrees of freedom and mean square terms. Next is R^2 and the coefficient of variation value. A R^2 value of 1 indicates a 'perfect' regression. The TEST OF OVERALL REGRESSION statement is a test of the hypothesis stated in Appendix B for a risk of ALPHA (α). If the FCAL exceeds F(ALPHA), H_0 is rejected and the alternative, H_1, is accepted. Under these conditions,

```
************************************************************************************************
************************************************************************************************
    FOLLOWING IS THE OUTPUT OF INPUT

    LINEAR=    0                              STAT=    1            ALPHA= 0.500000000D-01
        LINEAR= ZERO IMLIES NON-LINEAR            STAT= ONE IMPLIES POST STATISTICAL STUDY TO BE EXECUTED
        LINEAR= NON-ZERO IMPLIES LINEAR           STAT= ZERO IMPLIES NO POST SATISTICAL STUDY

    ACCURACY                         ACC= 0.10000D-05
    FOR GJWRLS                       CHECK= 0.10000D-19     DET=  1.0000
    FUDGE FACTOR FOR DTFIND          RF= 0.50000

    NUMBER OF DIFFERENTIAL EQ      NDE=       3
    NUMBER OF BOUNDARY CONDITIONS  NBV=       9
    NUMBER OF TERMS IN PWR SERIES  NTERMS=   20
    NO. EXACT BOUNDARY CONDITIONS  NEMAX=     1
    NO. OF NON-TRIVIAL DIFF EQ     NEQ=       1

    BOUNDARY VALUES     APPLIED ON(IBV)       IQBV      TIME OF BV
     117.30000               1                  1       0.0
     104.90000               1                  1        5.0000000
     95.300000               1                  1        10.000000
     87.300000               1                  1        15.000000
     79.200000               1                  1        20.000000
     71.900000               1                  1        25.000000
     65.700000               1                  1        30.000000
     59.800000               1                  1        35.000000
     54.000000               1                  1        40.000000

    INITIAL TIME OF INTEGRATION      TO= 0.0

    INITIAL GUESSES OF Y                     ICEX       LOWER LIMIT          UPPER LIMIT
         Y 1  YI( 1)=  110.00000              1          0.0                  0.0
         Y 2  YI( 2)= 0.40000000              1          0.0                  0.0
         Y 3  YI( 3)= 0.300000000D-01         1          0.0                  0.0

    PERTURBATION SCALER            PTBS= 0.100000000 00
        ABSOLUTE VALUE LIMITS      PTMIN= 0.0
                                   PTMAX=  2.0000

    IOUT=    0
    QUIT=   15

    CONCLUDES FORMAL OUTPUT OF THE INPUT
************************************************************************************************
************************************************************************************************
```

FIGURE 1

```
**********************************************************************************************************
**********************************************************************************************************
   FOLLOWING IS THE OUTPUT OF THE POST STATISTICAL STUDY

        SOURCE          SUM OF SQUARES         DEGREES OF FREEDOM        MEAN SQUARE
        ------          --------------         ------------------        -----------
        REGRESSION        63716.493                  3                    21238.831
        RESIDUAL          0.96692256                 6                    0.16115376      = S**2

        TOTAL(UNC)        63717.460                  9

   PER-CENT VARIATION    R**2=  99.973342          COEFFICIENT OF VARIATION   CV= 0.49129134D-02

   TEST OF OVERALL REGRESSION  FCAL=  131792.34      F(ALPHA)=  4.7571592       P(F(ALPHA).GT.FCAL)= 0.83446503D-06
        FOR RISK OF ALPHA= 0.50000000D-01
        *****ACCEPT REGRESSION*****

   SPECIFICS OF THE BOUNDARY CONDITIONS    T(ALPHA)=   2.4469080        ALPHA=  0.50000000D-01
        OBSERVED VALUES   ESTIMATED VALUE      RESIDUAL       ESTIMATED S.E.     LOWER LIMIT       UPPER LIMIT
          117.30000         116.89493        0.40507263        0.35231109        116.03285         117.75700
          104.90000         105.52164       -0.62164188        0.20683878        105.01553         106.02776
          95.300000         95.609284       -0.30928430        0.19612970        95.129373         96.089196
          87.300000         86.856482        0.44351834        0.20538341        86.353927         87.359036
          79.200000         79.037830        0.16217005        0.19521417        78.560159         79.515501
          71.900000         71.985065       -0.85064950D-01    0.17431129        71.558541         72.411589
          65.700000         65.568615        0.13138498        0.17133949        65.149363         65.987867
          59.800000         59.682176        0.11782363        0.21489984        59.156336         60.208016
          54.000000         54.240515       -0.24051461        0.30186196        53.501886         54.979143
        MEAN OF OBSERVATIONS=  81.711111
        SUM OF THE RESIDUALS= 0.34640960D-02

   SPECIFICS OF THE YI SOLUTIONS    T(ALPHA)=   2.4469080        ALPHA=  0.50000000D-01
        ESTIMATED VALUE        EST VARIANCE              CONFIDENCE INTERVAL
          116.89493             0.12412310           116.89493        (+-)  0.86207283
          0.50251918            0.79454589D-03       0.50251918       (+-)  0.68972685D-01
          0.16924118D-01        0.15727466D-05       0.16924118D-01   (+-)  0.30686476D-02

   COVARIANCE MATRIX OF THE SUPERPOSITION CONSTANTS
 0.31031D-01   0.61974D-01  -0.65375D-01
 0.61974D-01   0.31464      -0.40651
-0.65375D-01  -0.40651       0.54909

   CONCLUDES STAT PACKAGE
**********************************************************************************************************
**********************************************************************************************************
```

FIGURE 2

*****ACCEPT REGRESSION***** is printed. This is not to imply this is the 'best' possible regression, but for the stated hypothesis and the given data, it is an 'acceptable' regression.

Following the hypothesis test are two sections showing confidence intervals on the observations and state variables respectfully. These limits are all based on the same producer's risk (ALPHA) and t value (student's t distribution). In addition, the observed values (given boundary values) are shown with the estimated values from the regression. The residual terms;

$$\text{(observed value)} - \text{(estimated value)} \tag{5-4}$$

and the sum of the differences are included. Both sections contain the estimated variance, or standard error, of the individually estimated solutions.

The last section is the covariance matrix of the superposition constants (least squares). The diagonal elements are the variances of those constants.

A regression cannot be 'accepted' or 'rejected' on the basis of just one statistic. The programmer must qualify an acceptable solution and evaluate the overall regression, confidence limits, variances, hypothesis, etc. for the given system to determine whether a resulting model is acceptable. PSTAT does furnish enough information to evaluate the regression validity in most cases.

CHAPTER VI

CONCLUSIONS ON STATISTICAL RESULTS

PSTAT, a subroutine from the control program, provides a means of evaluating the acceptability of differential equation regressions. The control program and QUASII are based on the same theory and procedures (except for the integration methods). PSTAT is presently not directly compatible with QUASII, but does provide a basis on which QUASII solutions can be evaluated.

This statistical package can be used for linear and non-linear systems (see Reference 4 for types of non-linearities).

PSTAT cannot handle multiple observations for distinct value(s) of the independent variable(s). The ANOVA table and procedures should be modified to include pure error and further hypothesis testing.

The assumption of the linearity of equation 4-2 for the purpose of an analysis of variance, does not appear to be valid for examples worked with non-zero forcing functions (see equation 2-1). This area should be investigated from the possible approach of a non-linear analysis of variance.

In its present form, PSTAT coupled with QUASII provides a powerful tool for statistically evaluating many systems not previously modeled in a sufficient manner by standard algebraic regression packages.

REFERENCES

1) Bowker, Albert H. and Lieberman, Gerald J., Engineering Statistics. 2nd Edition, Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1972.

2) Boyce, William E. and DiPrima, Richard C., Elementary Differential Equations and Boundary Value Problems. 2nd Edition, New York: John Wiley & Sons, Inc., 1965.

3) Brunk, H.D., An Introduction to Mathematical Statistics. 2nd Edition, Waltham, Mass.: Blaisdell Publishing Company, 1965.

4) Childs, Bart, Porter, H.R., Excerpts from Numerical Solutions of Nonlinear Multipoint Boundary Value Problem. For the series Applied Mathematics and Computation. Addison Wesley Publishing Company.

5) Childs, Bart, Doiron, Harold, and Holloway, Carl, "Numerical Solutions of Multipoint Boundary Value Problems in Non-Linear Systems," Int. J. Systems Science, Vol. 2, No. 1, pg. 59-66, 1971.

6) Childs, Bart, Luckinbill, Dennis, Bryan, John, and Boyd, Jr., J.H., "Numerical Solutions of Multipoint Boundary Value Problems in Linear Systems" Int. J. Systems Science, Vol. 2, No. 1, pg. 49-57, 1971.

7) Draper, N.R. and Smith, H., Applied Regression Analysis, New York: John Wiley & Sons, Inc., 1966.

8) du Plessis, Roger M., "Poor Man's Explanation of Kalman Filtering or How I Stopped Worrying and Learned to Love Matrix Inversion," North American Aviation, Inc., Autonetics Division, Anahiem, California, 1967.

9) Freund, John E., Mathematical Statistics. 2nd Edition, Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1971.

10) Hald, A., Statistical Theory with Engineering Applications. New York: John Wiley & Sons, Inc., 1952.

11) Hamming, Richard W., Introduction to Applied Numerical Analysis. 1971.

12) Hicks, Charles R., Fundamental Concepts in the Design of Experiments. 2nd Edition, New York: Holt, Rinehart and Winston, 1973.

13) Hull, T.E., Enright, W.H., Fellen, B.M., and Sedgwich, A.E., "Comparing Numerical Methods for Ordinary Differential Equations," SIAM Journal of Numerical Analysis, Vol. 9, No. 4, pg. 603-637, 1972.

14) Lindgren, B.W. and McElrath, G.W., Introduction to Probability and Statistics. 2nd Edition, New York: The Macmillan Company, 1966.

15) Mann, Nancy R., Schafer, Ray E., and Singpurwalla, Nozer D., Methods for Statistical Analysis of Reliability and Life Data. New York: John Wiley & Sons, Inc., 1974.

16) Mood, Alexander M. and Graybill, Franklin A., Introduction to the Theory of Statistics. 2nd Edition, New York: McGraw-Hill Book Company, Inc., 1963.

17) Porter, III, H.R. and Childs, Bart, "Final Report for a Research Contract for the United States Government on Systems Identification."

18) Powers, David L., Boundary Value Problems. New York: Academic Press, 1972.

19) Stoer, Josef, "On the Numerical Solutions of Constrained Least-Squares Problems," SIAM Journal of Numerical Analysis, Vol. 8, No. 2, pg. 382-411, 1971.

20) Thomas, Jr., George B., Calculus and Analytic Geometry. 4th Edition, Menlo Park, California: Addison-Wesley Publishing Company, 1968.

21) Walker, III, John, "Fitting Criteria in Boundary Value Problems", ITC-02-08-76-009, I.En. 685 Report for Intern Training Center in conjunction with Texas A&M University.

22) Zupp, Jr., George Adam, "Identification of Systems by Quasilinearization," The University of Houston, 1968.

APPENDICES

APPENDIX A

NOMENCLATURE

A_f	Frontal area of a vehicle.
a	Superposition constant.
b	Boundary condition.
c	Coefficient of variation.
d	Transformation of boundary condition.
E	Expected value of.
f	Forcing function.
F	Theoretical value from Fisher's F Distribution Tables.
F_{cal}	Variable that is F distributed.
g	Acceleration of gravity.
H	Set of homogeneous solutions.
I	Identity matrix.
k	Number of boundary conditions to be met exactly.
L	Linear operator.
M	Mass.
m	Number of boundary conditions.
MS	Mean-square.
n	Order of the differential equation (number of state var.).
p	A particular solution.
P	Set of particular solutions.
q	Operator, either linear or non-linear.
r	Number of unknown initial conditions.
R^2	Percent variation due to regression.
s^2	Estimate of variance.

s'^2	Estimate of variance.
S	Matrix of the integrated values of the particular solutions.
SS	Sum of squares.
s.e.	Standard error of.
t	Independent variable time or a student's t statistic.
var	Variance of.
y	State variable.
μ	Coefficient of damping.
ξ	Spring constant.
W	Frequency.
α	Producer's risk.
β	Superposition constant.
μ_f	Rolling friction coefficient.
ρ	Air density.
C_d	Coefficient of drag.
ν	Degrees of freedom.
est.	Estimated.
e	Exact.
l	Least squares.
T	Transpose.
'	Transformation of by matrix operations.
^	Estimate.
—	Mean.

APPENDIX B

GENERAL LINEAR REGRESSION THEORY
(Draper and Smith, 1966)

To handle multiple linear regression, a matrix approach is desirable. Unless otherwise stated, all unsubscripted letters are matrices or vectors.

Consider a linear problem (only in the coefficients) with n observations and p independent variables, the following linear equation could arise;

$$z = W\beta + \varepsilon \tag{B-1}$$

where

z is a (n x 1) vector of observations

W is a (n x p) matrix of independent variable

β is a (n x 1) vector of parameters to be estimated

ε is a (n x 1) vector of errors

Certain basic assumptions must be made on the error terms, ε;

$$\begin{aligned}
&\varepsilon_i\text{'s are random variables} \\
&E(\varepsilon_i) = 0 \\
&\mathrm{var}(\varepsilon_i) = \sigma^2 \text{ (unknown)} \\
&\varepsilon_i \text{ and } \varepsilon_j \text{ are uncorrelated } (i \neq j) \quad \mathrm{cov}(\varepsilon_i, \varepsilon_j) = 0 \\
&\varepsilon_i\text{'s are normally distributed; } N(0, \sigma^2) \\
&\varepsilon_i\text{'s are therefore independent}
\end{aligned} \tag{B-2}$$

As a result of $\mathrm{cov}(\varepsilon_i, \varepsilon_j) = 0$, z_i and z_j are also uncorrelated and

$$E(z_i) = W\beta$$

$$\mathrm{var}(z_i) = \sigma^2$$

By evaluating the error sum of squares and substituting the least squares estimates of β (d an n vector), the following normal equation results:

$$(W^TW)^{-1}d = W^Tz \tag{B-3}$$

Using the decomposition of the total sum of squares (uncorrected), the ANOVA (analysis of variance) Table B.1 results.

TABLE B.1

Source	Sum of Squares	Degrees of Freedom	Mean Square
Due to regression	d^TW^Tz	p	MS_{REG}
About regression (residual)	$z^Tz - d^TW^T$	n-p	$MS_E = s^2$
Total (uncorrected)	z^Tz	n	

MS_E is an estimate of the variance about the regression.

Once the ANOVA table is constructed, several conclusions can be deduced:

I. A percent measurement of the total variation about the mean attributed to the regression (square of the multiple correlation coefficient) is given by;

$$R^2 = \frac{d^TW^Tz - n\bar{z}^2}{z^Tz - n\bar{z}^2} \tag{B-4}$$

where $\bar{z}$ is the mean of the observations.

A value of $R^2 = 1$ implies that the prediction is 'perfect'.

II. To test the overall regression equation, the hypothesis:

$$H_0: \beta_0 = \beta_1 = \ldots = \beta_p = 0 \tag{B-5}$$

must be tested against the alternate hypothesis:

$$H_1: \beta_i \neq 0 \quad i = 0 \ldots p \tag{B-6}$$

This employs the Fisher's F distribution. The F_{cal} statistic is:

$$F_{cal} = \frac{MS_{REG}}{MS_E} \tag{B-7}$$

This value is to be compared to a table value of;

$$F^{1-\alpha}_{(p,n-p)}$$

where:

$1-\alpha$ is a probability

p is the degree of freedom for the numerator

$n-p$ is the degree of freedom for the denominator

If F_{cal} exceeds the table value, the null hypothesis is rejected implying that the variation in the data is more than would be expected by chance in $(1-\alpha)100\%$ of similar data sets.

III. The covariance matrix of the least squares estimates (d) is;

$$\text{est. var}(d) = (W^T W)^{-1} s^2 \tag{B-8}$$

where the i^{th} diagonal elements is the estimated variance of d_i. Equation B-8 coupled with the student's t distribution yields a confidence interval for d_i with a risk of alpha (α) of:

$$d_i \pm t(n-p,\ 1-\alpha/2) \left[\text{estimate s.e. } (d_i)\right] \quad i=1\ldots n \tag{B-9}$$

IV. Similarly, a confidence interval for the estimates of z ($\hat{z}$) can be determined. Based on linear combinations of random variables, the estimated variance of $\hat{z}_i$ is;

$$\text{est. var}(\hat{z}_i) = W_i(W^TW)^{-1}W_i^Ts^2 \qquad i=1...n \qquad \text{(B-10)}$$

where W_i is the vector of W corresponding to the i^{th} row or boundary condition (least square). The resulting confidence interval is:

$$\hat{z}_i \pm t(n-p,\ 1-\alpha/2)\sqrt{\left[\text{est. var}(\hat{z}_i)\right]} \qquad i=1...n \qquad \text{(B-11)}$$

APPENDIX C

APPLICATIONS TO ALGEBRAIC REGRESSION

Illustration of the versatility of QUASII and verification of the control program can be shown utilizing the following algebraic regression:

$$x=b_0+b_1+b_2t^2+b_3t^3 \tag{C-1}$$

Equation C-1 is a linear third order regression equation.

Utilizing the following state variables:

$$\begin{aligned} y_1&=x=b_0+b_1t+b_2t^2+b_3t^3 \\ y_2&=\dot{x}=b_1+2b_2t+3b_3t^2 \\ y_3&=\ddot{x}=2b_2+6b_3t \\ y_4&=\dddot{x}=6b_3 \end{aligned} \tag{C-2}$$

Four differential equations result:

$$\begin{aligned} \dot{y}_1&=y_2 \\ \dot{y}_2&=y_3 \\ \dot{y}_3&=y_4 \\ \dot{y}_4&=0 \end{aligned} \tag{C-3}$$

Equations C-3 are linear with $\dot{y}_4$ being trivial.

By solving the set of differential equations C-3, the values of the y_i's at t=0 can be determined. Equation set C-2 yields the following results:

$$\begin{aligned} b_0&=y_1(0) \qquad & b_2&=y_3(0)/2 \\ b_1&=y_2(0) \qquad & b_3&=y_4(0)/6 \end{aligned} \tag{C-4}$$

or in the more general case:

$$b_n = y_{n+1}(0)/n! \tag{C-5}$$

APPENDIX D

THE CONTROL PROGRAM WITH OUTPUT OF THE NON-LINEAR MASS-SPRING-DAMPER SYSTEM

The control program is based on the same theory as QUASII and both are quite similar in structure. Several subroutines are common (with slight differences) to both programs. However, QUASII is capable of handling a larger variety of problems. As stated earlier, the main difference is that QUASII utilizes a 4^{th} order Runge-Kutta integrator opposed to power series integration employed by the control program. Because of this point a subroutine RECURL must be rewritten for each system to be modeled.

The following pages contain a copy of the control program main-line, supporting subroutines, and post-statistical package for the non-linear system of equation 2-1. The output does not include the results from the PSTAT package.

The program is well documented for both Case I and II problems.

APPENDIX D

THE CONTROL PROGRAM WITH OUTPUT OF THE NON-LINEAR MASS-SPRING-DAMPER SYSTEM

The control program is based on the same theory as QUASII and both are quite similar in structure. Several subroutines are common (with slight differences) to both programs. However, QUASII is capable of handling a larger variety of problems. As stated earlier, the main difference is that QUASII utilizes a 4^{th} order Runge-Kutta integrator opposed to power series integration employed by the control program. Because of this point a subroutine RECURL must be rewritten for each system to be modeled.

The following pages contain a copy of the control program main-line, supporting subroutines, and post-statistical package for the non-linear system of equation 2-1. The output does not include the results from the PSTAT package.

The program is well documented for both Case I and II problems.

```
                                H A S P   J O B   L O G

$15.48.29 JOB 3388 -- HUNTER   -- BEGINNING EXEC - INIT 5 - CLASS G - LEVEL 4
$15.52.02 JOB 3388 END EXECUTION.

*********************************************************************************

     USER ASSISTANCE SESSIONS ...
          THE NEXT SESSION, TO BE HELD ON TUESDAY, APRIL 6 AT 2 00
          IN THE DPC CONFERENCE ROOM, WILL BE DEVOTED TO COBOL USERS.
          EVERYONE IS ENCOURAGEC TO ATTEND.

     RECYCLE COMPUTER PAPER...
          CONTACT THE OCEANOGRAPHY GRADUATE COUNCIL AND YOUR OLD
          CCMPUTER PAPER WILL BE PICKED UP FOR RECYCLING.  FOR PICK UP,
          CALL 845-7211.

*********************************************************************************
---- HASP SPOOLING STATISTICS99999999999

   1,424 CARDS READ9999999999999999999

   1,836 LINES PRINTED99999999999999999

       0 CARDS PUNCHED99999999999999999
```

```
//HUNTER    JOB (S978,RRA,001,006,CH),'HUNTER'                              JOB3388
***PASSWORD     ********************************************************
***JOBPARM REGION=192


//  EXEC  FORTG,TIME=(,20),REGION=192K


XXGO        EXEC  PGM=FORTRANG,REGION=110K                                  00020000
***    COMPILER DD CARDS                                                    00040000
XXSYSPRT        DD  SYSOUT=A                                                00060000
XXSYSPUNCH      DD  SYSOUT=B                                                00080000
XXSYSLIN DD DSN=&&LOADSET,DISP=(MOD,DELETE),UNIT=SYSDA,                     00100000
XX  SPACE=(2240,(800,100),RLSE),DCB=(BLKSIZE=2240,BUFNO=3,OPTCD=C)          00120000
***    LOADER DD CARDS                                                      00140000
XX        DD  DDNAME=OBJECT          DEFINES HEX DECK FOR LOADER            00160000
//SYSLIB  DD
X/SYSLIB        DD  DSN=SYS1.FORTLIB,DISP=SHR                               00180000
//  DD
X/    DD  DSN=SYS1.DPCLIB,DISP=SHR                                          00200000
//  DD
X/  DD DSN=SYS1.SSPLIB,DISP=SHR                                             00220000
//  DD
X/        DD  DSN=USER.PLOTLIB,DISP=SHR                                     00240000
//  DD  DSN=SYSDPC.IMSLD.LOAD,DISP=SHR
//  DD  DSN=SYSDPC.IMSLS.LOAD,DISP=SHR
XXSYSLOUT  DD SYSOUT=A                                                      00260000
***    TAMU SUPPLIED GO DD CARDS                                            00280000
XXFT05F001 DD  DDNAME=SYSIN                                                 00300000
XXFT06F001 DD SYSOUT=A                                                      00320000
XXFT07F001 DD SYSOUT=B                                                      00340000
//SOURCE  DD  *
//SYSIN  DD  *
//
IEF236I ALLOC. FOR HUNTER   GO
IEF237I A06   ALLOCATED TO SYSPRT
IEF237I AA3   ALLOCATED TO SYSPUNCH
IEF237I 251   ALLOCATED TO SYSLIN
IEF237I 150   ALLOCATED TO SYSLIB
IEF237I 150   ALLOCATED TO
IEF237I 156   ALLOCATED TO
IEF237I 156   ALLOCATED TO
IEF237I 156   ALLOCATED TO
IEF237I 156   ALLOCATED TO
IEF237I A0C   ALLOCATED TO SYSLOUT
IEF237I A52   ALLOCATED TO FT05F001
IEF237I A10   ALLOCATED TO FT06F001
IEF237I AA4   ALLOCATED TO FT07F001
IEF237I A53   ALLOCATEC TO SOURCE
IEF142I - STEP WAS EXECUTED - COND CODE 0000 ---------------------------------- N O T E
IEF285I   SYS76097.T154830.RV000.HUNTER.LOADSET        DELETED
IEF285I   VOL SER NOS= WORK32.
IEF285I   SYS1.FORTLIB                                 KEPT
IEF285I   VOL SER NOS= VS2RES.
IEF285I   SYS1.DPCLIB                                  KEPT
IEF285I   VOL SER NOS= VS2RES.
IEF285I   SYS1.SSPLIB                                  KEPT
IEF285I   VOL SER NOS= USER02.
IEF285I   USER.PLOTLIB                                 KEPT
IEF285I   VOL SER NOS= USER02.
IEF285I   SYSDPC.IMSLD.LOAD                            KEPT
```

```
IEF285I   VOL SER NOS= USER02.
IEF285I   SYSOPC.IMSLS.LOAD                          KEPT
IEF285I   VOL SER NOS= USER02.
TAM001I STEP /GO/ REGION SIZE = 192K - CORE USED = 192K

TAM002I STEP /GO/ EXEC TIME = 5.14 SEC - I/O TIME = 1.98 SEC
TAM003I STEP /GO/ STEP SETUP = 1.67 SEC - I/O OVERHEAD = .00 SEC
TAM004I STEP /GO/ PAGES IN = 452 - PAGES OUT = 148
```

```
C-----------------------------------------------------------------------MAIN0000
C-----------------------------------------------------------------------MAIN0010
C     THIS PROGRAM IS A NON-LINEAR REGRESSION PACKAGE WHICH IS          MAIN0020
C     CAPABLE OF REGRESSION ON EITHER 10TH ORDER DIFFERENTIAL           MAIN0030
C     OR 9TH ORDER ALGEBRARIC EQUATIONS. POWER SERIES INTEGRATION       MAIN0040
C     IS USED. ALSO AVAILABLE IS AN ANALYSIS OF VARIANCE OF THE RESULT  MAIN0050
C-----------------------------------------------------------------------MAIN0060
C-----------------------------------------------------------------------MAIN0070
      IMPLICIT REAL*8(A-H,O-Z)                                          MAIN0080
      REAL*8          YI(10),P(10,11),PC(20,10,11),TBV(25)              MAIN0090
      REAL*8          DU(20,20),S(20,20),BV(25),PDOT(11)                MAIN0100
      REAL*8 C(10),PTB(10),ULIM(10),LLIM(10)                            MAIN0110
      REAL*8 W(25,10),ZHAT(25,1),ZEE(25,1),PB(10),YIR(10)               MAIN0120
      REAL*8 ALS(10,1),DLS(25,1)                                        MAIN0130
      INTEGER IBV(25),IQBV(25),ICEX(10),QUIT,STAT                       MAIN0140
      COMMON/REG/ZEE,ZHAT,W,PB,YIR,ALS,DLS,ZPAR                         MAIN0150
      COMMON/PARM/TO,TC,NDE,NP,NBV,NTERMS,LINEAR                        MAIN0160
      COMMON/PYPS/YI,P,PC,TBV                                           MAIN0170
      COMMON/LOG/INPT,IPRT                                              MAIN0180
      DATA S/400*0.0D0/                                                 MAIN0190
      DATA C/10*0.0D0/                                                  MAIN0200
      DATA PTB/10*0.0D0/                                                MAIN0210
      DATA ULIM/10*0.0D0/                                               MAIN0220
      DATA LLIM/10*0.0D0/                                               MAIN0230
      DATA STAT/0/                                                      MAIN0240
      DATA IBV/25*0/                                                    MAIN0250
      DATA IQBV/25*0/                                                   MAIN0260
      DATA ICEX/10*0/                                                   MAIN0270
C-----------------------------------------------------------------------MAIN0280
C     ITER IS THE NUMBER OF THE ITERATION BEING PERFORMED               MAIN0290
C     ACC IS ACCURACY FACTOR FOR DTFIND, EVAL, DEVAL                    MAIN0300
C     CHECK ACCURACY FACTOR IN GJRWLS                                   MAIN0310
C     DET IS AN INITIAL VALUE FOR DETERMINATE IN GJRWLS                 MAIN0320
C     RF IS FUDGE FACTOR FOR DTFIND, WILL USUALLY BE 1                  MAIN0330
C     LINEAR = 0 IMPLIES NON-LINEAR SYSTEM                              MAIN0340
C     LINEAR NOT EQUAL TO 0 IMPLIES LINEAR SYSTEM                       MAIN0350
C     STAT IS A FLAG FOR THE POST STATISTICAL PACKAGE(PSTAT)            MAIN0360
C      EQUALS 1 IMPLIES STAT PAK EXECUTED                               MAIN0370
C      EQUALS 0 IMPLIES NO STAT PAK                                     MAIN0380
C     ALPHA IS THE CONSUMERS RISK THAT IS FOR A CONFIDENCE              MAIN0390
C      OF 95 , ALPHA WOULD EQUAL 0.05                                   MAIN0400
C     NDE IS THE NUMBER OF DIFF EQ                                      MAIN0410
C     NBV IS THE NUMBER OF BOUNDARY VALUES                              MAIN0420
C     NTERMS IS THE NUMBER OF TERMS IN THE POWER SERIES TO BE CALCULATED MAIN0430
C     NEMAX IS THE NUMBER OF BOUNDARY VALUES TO BE MET EXACTLY          MAIN0440
C      NEMAX MUST BE AT LEAST 1,BECAUSE THE SUMMATION OF THE            MAIN0450
C      SUPERPOSITION CONSTANTS MUST EQUAL 1. THERFFORE IT WILL          MAIN0460
C      BE EQUAL TO THE NUMBER OF EXACT BOUNDARY CONDITIONS PLUS 1       MAIN0470
C     NE- IS THE NUMBER OF NON-TRIVIAL DIFFERENTIAL EQUATION            MAIN0480
C      THUS THE NUMBER OF TIME DEPENDENT VARIABLES                      MAIN0490
C     TBV IS TIME OF THE BOUNDARY VALUE                                 MAIN0500
C     BV IS THE BOUNDARY VALUE                                          MAIN0510
C     IBV IS THE DESIGNATION FOR WHICH Y THE BOUNDARY COND IS ON        MAIN0520
C      A NEGATIVE VALUE INDICATES THE DERIVATIVE OF THAT VARIABLE       MAIN0530
C     IQBV=0 IMPLIES BOUNDARY CONDITION TO BE MET EXACTLY               MAIN0540
C     IQBV=NON-ZERO IMPLIES BOUNDARY CONDITIONS IN LEAST SQARES SINCE   MAIN0550
C     TO IS THE INITIAL TIME OF THE POWER SERIES                        MAIN0560
C-----------------------------------------------------------------------MAIN0570
```

```
0021            INPT=5
          C------------------------------------------------------------------
          C   YI ARE THE INITIAL GUESSES OF THE Y VALUES
          C    THE FIRST NEQ VARIABLES ARE TIME DEPENDENT
          C    THE REMAINING NDE-NEQ VARIABLES ARE NON-TIME DEPENDENT CONSTANTS
          C   ICEX GREATER THAN 0 IMPLIES NOT EXACT KNOWN INITIAL CONDITION
          C   ICEX EQUAL TO ZERO IMPLIES THAT THE INITIAL CONDITION
          C    HAS AN UPPER AND LOWER BOUND(LLIM AND ULIM)
          C   ICEX LESS THAN 0  IMPLIES EXACT INITIAL CONDITION
          C    IF A VARIABLE HAS AN ICEX LESS THAN ZERO, THUS EXACT KNOWN
          C    INITIAL CONDITION  THE YI GUESS SHOULD BE THAT EXACT KNOWN
          C    INITIAL CONDITION  ALSO, IF KNOWN EXACTLY THERE CANNOT BE ANY
          C    BOUNDARY CONDITIONS ON THIS VARIABLE MET EXACTLY
          C    IF IT IS NON-TIME DEPENDENT, A CONSTANT
          C   LLIM IS A LOWER BOUND ON THE INITIAL CONDITION
          C   ULIM IS AN UPPER BOUND ON THE INITIAL CONDITION
          C    LLIM AND ULIM ONLY APPLY IF ICEX=0
          C   PTBS IS A PERTURBATION SCALER TO DETERMINE THE SIZE OF PTB(I)
          C    PTMIN IS THE MIN VALUE OF THE ABSOLUTE VALUE OF THE PERTURBATION
          C    PTMAX IS THE MAX VALUE OF THE ABSOLUTE VALUE OF THE PERTURBATION
          C------------------------------------------------------------------
0022            IPRT=6
          C------------------------------------------------------------------
          C   ICUT IS AN OUTPUT-DEBUG VARIABLE
          C    .EQ. -1 IMPLIES OUTPUT OF PSTAT PACKAGE
          C    .EQ. 0 IMPLIES NO DEBUG OPTIONS IN AFFECT
          C    .GT. 1 IMPLIES PIVOT VALUES FROM GJRWLS
          C    .GT. 3 IMPLIES P MATRIX BEFOER CALCR
          C    .EQ. 4 IMPLIES PARTICULAR SOLUTIONS FROM CALCSR**VOLUMINOUS**
          C    .GT. 5 IMPLIES OUTPUT FROM EVAL OR DEVAL
          C   QUIT IS A VARIABLE WHICH WILL TERMINATE THE PROGRAM
          C    BEFORE THE OPTIMAL SOLUTION IS DETERMINED. THE REASON
          C    FOR THIS IS TO BE ABLE TO RUN LESS THAN THE MINIMUM
          C    TIME IF THE PROGRAM CONVERGES SLOWLY.
          C    QUIT THEREFORE SPECIFIES THE NUMBER OF ITERATIONS
          C    IF SPECIFIED ZERO, IT HAS NO AFFECT ON THE PROGRAM
          C------------------------------------------------------------------
          C     INPUT
          C
0023       1    CONTINUE
0024            ITER=1
0025            NOSTT=0
0026            ACC=1.0D-06
0027            CHECK=1.0D-20
0028            DET=1.0D0
0029            RF=0.5D0
0030            READ(INPT,600,END=520)LINEAR,STAT,ALPHA
0031     600    FORMAT(2I5,G10.0)
0032            READ(INPT,610)NDE,NBV,NTERMS,NEMAX,NEQ
0033     610    FORMAT(5I5)
0034            READ(INPT,620)(TBV(I),BV(I),IBV(I),IQBV(I),I=1,NBV)
0035     620    FORMAT(2G10.0,10X,2I5)
0036            READ(INPT,630)TO
0037     630    FORMAT(G10.0)
0038            READ(INPT,640)(YI(I),ICEX(I),LLIM(I),ULIM(I),I=1,NDE)
0039     640    FORMAT(G10.0,5X,I5,5X,G10.0,5X,G10.0)
0040            READ(INPT,650)PTBS,PTMIN,PTMAX
0041     650    FORMAT(3G10.0)
```

```
0042              READ(INPT,660)IOUT,QUIT
0043        660   FORMAT(I5,I5)
            C
            C     CONCLUDES INPUT
            C-----------------------------------------------------------------
            C     OUTPUT OF THE INITIAL INPUT
            C
0044        665   CONTINUE
0045              WRITE(IPRT,670)
0046        670   FORMAT(1H1,2(110(1H*)/),6X, 32HFOLLOWING IS THE OUTPUT OF INPUT)
0047              WRITE(IPRT,680)LINEAR,STAT,ALPHA
0048        680   FORMAT(1H0,5X,7HLINEAR=,I5,30X,5HSTAT=I5,10X,6HALPHA=,G15.8,/,
                 A10X,30HLINEAR= ZERO IMLIES NON-LINEAR,12X,
                 B55HSTAT= ONE IMPLIES POST STATISTICAL STUDY TO BE EXECUTED,/,
                 C10X,31HLINEAR= NON-ZERO IMPLIES LINEAR,11X,
                 D43HSTAT= ZERO IMPLIES NO POST SATISTICAL STUDY)
0049              WRITE(IPRT,690)ACC,CHECK,DET,RF
0050        690   FORMAT(1H0,5X,8HACCURACY,23X,4HACC=,G12.5,/,
                 A6X,10HFOR GJWRLS,21X,6HCHECK=,G12.5,5X,4HDET=G12.5,/,
                 B6X,23HFUDGE FACTOR FOR DIFIND,8X,3HRF=,G12.5)
0051              WRITE(IPRT,700)NDE,NBV,NTERMS,NEMAX,NEQ
0052        700   FORMAT(1H0,5X,25HNUMBER OF DIFFERENTIAL EQ, 6X,4HNDE=,3X,I5,/
                 16X,29HNUMBER OF BOUNDARY CONDITIONS,2X,4HNBV=,3X,I5,/,
                 26X,29HNUMBER OF TERMS IN PWR SERIES,2X,7HNTERMS=,I5/,
                 36X,29HNO. EXACT BOUNDARY CONDITIONS,2X,6HNEMAX=,1X,I5,/,
                 46X,26HNO. OF NON-TRIVIAL DIFF EQ,5X,4HNEQ=,3X,I5)
0053              WRITE(IPRT,710)(BV(J),IBV(J),IQBV(J),TBV(J),J=1,NBV)
0054        710   FORMAT(1H0,5X,15HBOUNDARY VALUES,5X,15HAPPLIED ON.IBV),6X,
                 16H IQBV ,5X,10HTIME OF BV,
                 225(/,5X,G15.8,10X,I5,10X,I5,07X,G15.8))
0055              WRITE(IPRT,720)TO
0056        720   FORMAT(1H ,5X,27HINITIAL TIME OF INTEGRATION,6X,3HTO=,G15.8,/
                 1/,6X,20HINITIAL GUESSES OF Y,20X,4HICEX,
                 208X,11HLOWER LIMIT,10X,11HUPPER LIMIT)
0057              WRITE(IPRT,730)(J,J,YI(J),ICEX(J),LLIM(J),ULIM(J),J=1,NDE)
0058        730   FORMAT(1H ,10X,1HY,I2,2X,3HYI(,I2,2H)=,G15.8,5X,I5,
                 210X,G15.8,6X,G15.8)
0059              WRITE(IPRT,740)PTBS,PTMIN,PTMAX
0060        740   FORMAT(1H0,5X,19HPERTURBATION SCALER,10X,5HPTBS=,G15.8,/,
                 A10X,21HABSOLUTE VALUE LIMITS,4X,6HPTMIN=,G12.5,/,
                 B35X,6HPTMAX=,G12.5)
0061              WRITE(IPRT,750)IOUT,QUIT
0062        750   FORMAT(1H0,5X,5HIOUT=,I5,/,6X,5HQUIT=,I5)
0063              WRITE(IPRT,760)
0064        760   FORMAT(1H0,5X,36HCONCLUDES FORMAL OUTPUT OF THE INPUT,
                 A2(/,110(1H*)))
            C
            C     CONCLUDES THE OUTPUT OF THE INPUT
            C-----------------------------------------------------------------
            C-----------------------------------------------------------------
            C   THIS LOOP PROVIDES A SAFEGUARD ON LINEAR INDEPENDENCE OF CONSTRAINTS
            C    IF A NON-TIME DEPENDENT VARIABLE IS TO BE MET EXACTLY IN A
            C    BOUNDARY CONDITION, THIS CAUSES THE BOUNDARY CONDITION
            C    TO BECOME AN EXACT KNOWN INITIAL CONDITION AND THAT THE
            C    BOUNDARY CONDITION WOULD ALSO BE MET IN A LEAST SQUARE SINCE
            C
0065              IC=0
```

```
0067              IF(IBV(I).LE.NEQ.OR.IQBV(I).NE.0) GO TO 5
0068          NEMAX=NEMAX-1
0069          ID=IBV(I)
0070          YI(ID)=BV(I)
0071          IQBV(I)=1
0072          ICEX(ID)=-1
0073          WRITE(IPRT,770)ID,YI(ID),I,ICEX(ID),IQBV(I),NEMAX
0074      770 FORMAT(1H0,5X,38HCORRECTION DUE TO EXACT BC ON CONSTANT,/
         110X,11HAFFECTS YI(,I2,1H),5X,10HNOW ECUALS,2X,G15.8,/,
         210X,27HFROM BCUNDARY CONDITION NO.,I5,/,
         326X,5HICEX=,I5,/,26X,5HIQBV=,I5,/,26X,6HNEMAX=,I5,/,
         46X,37HTHEREFORE REPRINT THE CORRECTED INPUT,/,2(110(1H*)/))
0075      5   CONTINUE
0076              IF(ID.NE.0) GO TO 665
        C-----------------------------------------------------------------
        C-----------------------------------------------------------------
        C    IF THE SOLUTIONS HAVE NOT MET THE ACCURACY SPECIFIED BY ACC
        C     THE PROGRAM RETURNS HERE AND STARTS AGAIN
        C
0077      9876 CONTINUE
        C-----------------------------------------------------------------
        C    NCNT IS JUST A COUNTER
        C    NCNTEX IS NCNT FOR BOUNDARY CONDITIONS TO BE MET EXACTLY
        C      EXACT CONDITIONS GO IN FIRST ROWS OF S MATRIX
        C    NCNTLS IS NCNT FOR BOUNDARY CONDITIONS TO BE MET IN LEAST SQUARES
        C      SINCE, THESE OCCUPY ROWS AFTER EXACT CONDITIONS IN S MATRIX
        C    NBVAT IS THE DESIGNATION OF THE BOUNDARY VALUE YOU ARE AT
        C    NROWA,NRDU,NRS ARE ALL NUMBER OF ROWS IN APPROPRIATE MATRIX
        C
0078          NCNT=1
0079          NCNTEX=1
0080          NCNTLS=NEMAX
0081          NBVAT=1
0082          NRS=NBV+1
0083          NRDU=NBV+1
0084          NROWA=NBV+1
0085          TC=TO
        C-----------------------------------------------------------------
        C    PLACES INITIAL GUESSES AND PERTURBED ONES IN THE P MATRIX
        C    LOOP 10 SETS FIRST COLUMN OF P MATRIX EQUAL TO
        C    UNPERTURBED INITIAL GUESSES OF YI
        C    NP IS THE NUMBER OF PARTICULAR SOLUTIONS
        C     NP IS EQUAL TO THE NUMBER OF DIFFERENTIAL EQUATIONS (VARIABLES)
        C     PLUS ONE MINUS THE NUMBER OF EXACT KNOWN INITIAL CONDITIONS
        C
0086          NP=1
0087          DO 10 I=1,NDE
0088          P(I,1)=YI(I)
        C    THE FOLLOWING IF STATEMENT ALLOWS FOR THE PERTURBATION
        C     SIZE TO DEFAULT TO ZERO IF THE VARIABLE HAS AN EXACT
        C     KNOWN INITIAL CONDITION
0089              IF(ICEX(I).LT.0) GO TO 8
0090          NP=NP+1
        C    THE PERTURBATION WILL BE (PTBS*100)  OF THE INITIAL YI(I) GUESS
        C     THE ABSOLUTE VALUE OF PTB(I) WILL NOT BE LESS THAN PTMIN
        C     OR GREATER THAN PTMAX
0091          PTB(I)=PTBS*YI(I)
0092              IF(DABS(PTB(I)).LT.PTMIN) PTB(I)=DSIGN(PTMIN,PTB(I))
```

```
0093             IF(DABS(PTB(I)).GT.PTMAX) PTB(I)=DSIGN(PTMAX,PTB(I))          MAIN2320
0094         GO TO 10                                                          MAIN2330
0095      8  CONTINUE                                                          MAIN2340
0096         PTB(I)=0.0D0                                                      MAIN2350
0097     10  CONTINUE                                                          MAIN2360
        C----------------------------------------------------------------------MAIN2370
        C  NUNK IS THE NUMBER OF UNKNOWN VARIABLES                             MAIN2380
0098         NUNK=NP-1                                                         MAIN2390
0099         IP=0                                                              MAIN2400
        C                                                                      MAIN2410
        C  LOOP 25 SETS IN THE REST OF THE PERTURBED VALUES OF YI INTO         MAIN2420
        C  THE P MATRIX.  ALSO, IT TAKES INTO  ACCCUNT KNOWN INITIAL COND      MAIN2430
        C                                                                      MAIN2440
0100         DO 25 I=1,NUNK                                                    MAIN2450
0101     15  IP=IP+1                                                           MAIN2460
0102         IF(ICEX(IP).LT.0)GO TO 15                                         MAIN2470
0103           DO 20 J=1,NDE                                                   MAIN2480
0104           P(J,I+1)=YI(J)                                                  MAIN2490
0105     20    CONTINUE                                                        MAIN2500
0106         P(IP,I+1)=P(IP,I+1)+PTB(IP)                                       MAIN2510
0107     25  CONTINUE                                                          MAIN2520
        C                                                                      MAIN2530
        CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDMAIN2540
0108           IF(IOUT.LT.3.OR.IQUT.GE.QUIT) GO TO 35                          MAIN2550
0109           WRITE(IPRT,780)                                                 MAIN2560
0110    780    FORMAT(1H1,22HP MATRIX BEFORE CALCSR,/)                         MAIN2570
0111                DO 30 I=1,NDE                                              MAIN2580
0112                WRITE(IPRT,790)(P(I,J),J=1,NP)                             MAIN2590
0113     30         CONTINUE                                                   MAIN2600
0114    790         FORMAT(1H ,10G12.5)                                        MAIN2610
0115     35    CONTINUE                                                        MAIN2620
        CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDMAIN2630
        C                                                                      MAIN2640
        C  THIS LOOP SETS FIRST ROW OF S MATRIX =1 BECAUSE                     MAIN2650
        C  SUMMATION OF A'S MUST BE 1.                                         MAIN2660
        C                                                                      MAIN2670
0116                NX=NP+1                                                    MAIN2680
0117                DO 40 JA=1,NX                                              MAIN2690
0118                S(1,JA)=1.0D00                                             MAIN2700
0119     40         CONTINUE                                                   MAIN2710
        C----------------------------------------------------------------------MAIN2720
        C  CALCULATES THE SERIES AT CENTER OF EXPANSION                        MAIN2730
0120     45  CALL CALCSR(ICUT)                                                 MAIN2740
        C                                                                      MAIN2750
        C  CALCULATES THE SMALLEST STEP SIZE POSSIBLE OVER ENTIRE RANGE        MAIN2760
0121         DI=TBV(NBV)-TC                                                    MAIN2770
0122         DT=DI                                                             MAIN2780
0123                DO 50 K=1,NP                                               MAIN2790
0124                CALL DTFIND(PC(1,1,K),NEQ,NTERMS,RF,ACC,DI,DTI)            MAIN2800
0125                IF(DTI.LT.DT) DT=DTI                                       MAIN2810
0126     50         CONTINUE                                                   MAIN2820
        C                                                                      MAIN2830
        C  DTS IS THE DESIRED STEP SIZE TO THE NEXT BOUNDARY CONDITION         MAIN2840
        C  DTE IS THE DESIRED STEP SIZE TO THE NEXT BOUNDARY CONDITION         MAIN2850
0127         DTS=TBV(NBVAT)-TC                                                 MAIN2860
0128     55  DTE=TBV(NBVAT)-TC                                                 MAIN2870
        C  IF DT GREATER THAN DTE THEN DESIRED STEP CAN BE TAKEN               MAIN2880
```

```
          C   IF DT LESS THAN DTE, CENTER OF EXPANSION IS INCREASED BY DT          MAIN2900
 0130           TC=TC+DT                                                           MAIN2910
          C                                                                        MAIN2920
          C   SERIES EVALUATED AT NEW CENTER OF EXPANSION                          MAIN2930
 0131                     DO 60 I=1,NP                                             MAIN2940
 0132                     CALL EVAL(PC(1,1,I),P(1,I),NEQ,NTERMS,ACC,DT)            MAIN2950
 0133      60             CONTINUE                                                 MAIN2960
          C                                                                        MAIN2970
          CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDMAIN2980
 0134               IF(TC.LT.TBV(NBVAT)) GO TO 70                                  MAIN2990
 0135               IF(IOUT.LT.5.OR.IOUT.GE.QUIT) GO TO 70                         MAIN3000
 0136               WRITE(IPRT,800)DT                                              MAIN3010
 0137      800      FORMAT(1H1,18HFROM LOOP 60  EVAL,5X,3HDT=,G15.8,/)             MAIN3020
 0138                     DO 65 I=1,NDE                                            MAIN3030
 0139                     WRITE(IPRT,810)(P(I,J),J=1,NP)                           MAIN3040
 0140      65             CONTINUE                                                 MAIN3050
 0141      810            FORMAT(2H ,10G12.5)                                      MAIN3060
 0142      70       CONTINUE                                                       MAIN3070
          CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCMAIN3080
          C                                                                        MAIN3090
          C   NOW TO RECALCULATE THE SERIES                                        MAIN3100
 0143          GO TO 45                                                            MAIN3110
          C------------------------------------------------------------------------MAIN3120
          C------------------------------------------------------------------------MAIN3130
          C   75 ON IMPLIES THAT THE DESIRED STEP WAS SMALL ENOUGH AT ABOUT 55     MAIN3140
 0144      75             DO 80 I=1,NP                                             MAIN3150
 0145                     CALL EVAL(PC(1,1,I),P(1,I),NEQ,NTERMS,ACC,DTE)           MAIN3160
 0146      80             CONTINUE                                                 MAIN3170
          C                                                                        MAIN3180
          CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDMAIN3190
 0147               IF(TC.LT.TBV(NBVAT)) GO TO 90                                  MAIN3200
 0148               IF(IOUT.LT.5.OR.IOUT.GE.QUIT) GO TO 90                         MAIN3210
 0149               WRITE(IPRT,820)DTE                                             MAIN3220
 0150      820      FORMAT(1H1,18HFROM LOOP 80  EVAL,5X,4HDTE=,G15.8,/)            MAIN3230
 0151                     DO 85 I=1,NDE                                            MAIN3240
 0152                     WRITE(IPRT,810)(P(I,J),J=1,NP)                           MAIN3250
 0153      85             CONTINUE                                                 MAIN3260
 0154      90       CONTINUE                                                       MAIN3270
          CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDMAIN3280
          C                                                                        MAIN3290
          C------------------------------------------------------------------------MAIN3300
          C   THROUGH 100 MAKES SURE THAT EXACT BOUNDARY VALUES ARE PLACED IN      MAIN3310
          C     THE BEGINNING OF THE S MATRIX AND THE LEAST SQUARE CONDITIONS      MAIN3320
          C     IN THE FOLLOWING ROWS                                              MAIN3330
          C                                                                        MAIN3340
 0155          IJK=IBV(NBVAT)                                                      MAIN3350
 0156          IF(IQBV(NBVAT).NE.0) GO TO 95                                       MAIN3360
 0157          NCNTEX=NCNTEX+1                                                     MAIN3370
 0158          NCNT=NCNTEX                                                         MAIN3380
 0159          GO TO 100                                                           MAIN3390
 0160      95  NCNTLS=NCNTLS+1                                                     MAIN3400
 0161          NCNT=NCNTLS                                                         MAIN3410
 0162      100 CONTINUE                                                            MAIN3420
          C                                                                        MAIN3430
          C   120 LOOP IS ESTABLISHING THE S MATRIX                                MAIN3440
 0163           IDOT=0                                                             MAIN3450
 0164                     DO 120 JA=1,NP                                           MAIN3460
 0165                           IF(IJK.LE.0) GO TO 105                             MAIN3470
```

```
                  TEMP=P(IJK,JA)                                           MAIN3480
                  GO TO 120                                                MAIN3490
      105              IF(IJK.LT.0) GO TO 115                              MAIN3500
                  TEMP=0.0D0                                               MAIN3510
                       DO 110 IJ=1,NDE                                     MAIN3520
      110              TEMP=TEMP+C(IJ)*P(IJ,JA)                            MAIN3530
                  GO TO 120                                                MAIN3540
      115         CONTINUE                                                 MAIN3550
                  CALL DEVAL(PC(1,1,JA),PDOT(1),NEQ,NTERM,ACC,DTE)         MAIN3560
                  IDOT=1                                                   MAIN3570
                  IJJ=IABS(IJK)                                            MAIN3580
                  TEMP=PDOT(IJJ)                                           MAIN3590
      120         S(NCNT,JA)=TEMP                                          MAIN3600
C                                                                          MAIN3610
CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDMAIN3620
            IF(IDOT.EQ.0) GO TO 130                                        MAIN3630
            IF(IOUT.LT.5.OR.IQUT.GE.QUIT) GO TO 130                        MAIN3640
            IF(TC.LT.TBV(NBVAT)) GO TO 130                                 MAIN3650
            WRITE(IPRT,830)DTE                                             MAIN3660
      830   FORMAT(1H1,10HFROM DEVAL,5X,4HDTE=,G15.8,/)                    MAIN3670
                  DO 125 I=1,NDE                                           MAIN3680
                  WRITE(IPRT,810)(P(I,J),J=1,NP)                           MAIN3690
      125         CONTINUE                                                 MAIN3700
      130   CONTINUE                                                       MAIN3710
CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDMAIN3720
C                                                                          MAIN3730
C    ESTABLISH LAST COLUMN OF S MATRIX AS BOUNDARY VALUES                  MAIN3740
        S(NCNT,NP+1)=BV(NBVAT)                                             MAIN3750
C                                                                          MAIN3760
        NBVAT=NBVAT+1                                                      MAIN3770
C    IF ALL BV HAVE NOT BEEN USED GO TO 200 AND CHECK ON STEP INCREMENT    MAIN3780
C     THEN GO TO 45                                                        MAIN3790
        IF (NBVAT.LE.NBV) GO TO 200                                        MAIN3800
C                                                                          MAIN3810
CPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPMAIN3820
C    THIS SECTION ESTABLISHES A WORKING VEXTOR OF THE                      MAIN3830
C     BOUNDARY VALUES(ZEE) AND THE CALCULATED VALUES(ZHAT).                MAIN3840
C     ALSO THE AVERAGE OF ZEE IS CALCULATED,ZBAR                           MAIN3850
C     ONLY THOSE MET IN A LEAST SQUARES SINCE ARE CONSIDERED               MAIN3860
C                                                                          MAIN3870
        IF(STAT.EQ.0.OR.NDSTT.NE.1) GO TO 145                              MAIN3880
        NLS=NEMAX+1                                                        MAIN3890
              DO 135 J=NLS,NRS                                             MAIN3900
              JJ=J-NEMAX                                                   MAIN3910
              ZHAT(JJ,1)=S(J,1)                                            MAIN3920
      135     CONTINUE                                                     MAIN3930
        ZBAR=0.0D0                                                         MAIN3940
        II=1                                                               MAIN3950
              DO 140 I=1,NBV                                               MAIN3960
              IF(IOBV(I).EQ.0) GO TO 140                                   MAIN3970
              ZEE(II,1)=BV(I)                                              MAIN3980
              ZBAR=ZBAR+ZEE(II,1)                                          MAIN3990
              II=II+1                                                      MAIN4000
      140     CONTINUE                                                     MAIN4010
C    NRZ IS THE NUMBER OF LEAST SQUARE BV                                  MAIN4020
C    NRW IS THE NUMBER OF ROWS OF THE INDEPENDENT VARIABLE                 MAIN4030
        NRZ=II-1                                                           MAIN4040
```

```
0207          ZBAR=ZBAR/DFLOAT(NRZ)
0208          GO TO 230
0209     145  CONTINUE
        C
        C   ESTABLISHES VECTORS OF THE YI SOLUTION AND THEIR ASSOCIATED .
        C    PERTURBATIONS. ONLY THOSE NOT KNOWN EXACTLY ARE CONSIDERED
0210          IF(STAT.EQ.0) GO TO 152
0211          II=0
0212               DO 150 I=1,NDE
0213               IF(ICEX(I).LT.0) GO TO 150
0214               II=II+1
0215               PB(II)=PTB(I)
0216               YIR(II)=YI(I)
0217     150       CONTINUE
0218     152  CONTINUE
        CPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
        C
        C-----------------------------------------------------------------------
        C   OUTPUT THE S MATRIX
0219          WRITE(IPRT,840)ITER
0220     840  FORMAT(1H1,5X,9HITERATION,I5)
0221          WRITE(IPRT,850)
0222     850  FORMAT(1H0,5X,48HS MATRIX(CONTAINS UNPERTERBED AND PERTURBED SOL))
0223                  JEND=NP+1
0224                  DO 155 JBOB=1,NRS
0225     155          WRITE(IPRT,860)(S(JBOB,JBIG),JBIG=1,JEND)
0226     860          FORMAT(10X,9(1X,G12.5)) .
0227          ITER=ITER+1
        C-----------------------------------------------------------------------
        C   GUASS-JORDAN REDUCTION WITH LEAST SQUARES
        C
0228           CALL GJRWLS
             1(S,DU,NRS,NRDU,NEMAX,DET,CHECK,IOUT,NP,NROWA,IPUNT,IDUM,
             2STAT,W,DLS)
        C
        C-----------------------------------------------------------------------
        C   FOLLOWING GIVES DESIRED OUTPUT
        C   OUTPUTS SUPERPOSITIONING CONSTANTS
        C
0229          WRITE(IPRT,870)
0230     870  FORMAT(1H0,5X,23HSUPERPOSITION CONSTANTS)
0231             DO 160 NT=1,NP
0232     160     WRITE(IPRT,880) NT,SINT,NX)
0233     880     FORMAT(1H ,10X,2HA(,I2,4H) = ,G15.8)
        C
        C   OUTPUT THE VALUES OF Y           .
        C   ALSO DETERMINES THE NEW PERTURBED VALUES OF YI
0234          WRITE(IPRT,885)
0235     885  FORMAT(1H0,5X,  3 SOLUTION FOR STATE VARIABLES)
0236          ICQ=1
0237            DO 165 IQ=1,NDE
0238            ICQ=ICQ+1
0239            YI(IQ)=YI(IQ)+S(ICQ,NX)*PTB(IQ)
0240            WRITE(IPRT,890) IQ,YI(IQ)
0241                 IF(ICEX(IQ).NE.0) GO TO 165
0242                 IF(YI(IQ).GT.ULIM(IQ)) YI(IQ)=ULIM(IQ)
```

```
0245          890  FORMAT(1H ,10X,13HTRUE IC OF Y(,I2,5H) IS ,G15.8)
              C
              CPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
0246               IF(STAT.NE.1) GO TO 175
0247                    DO 170 I=2,NP
0248                    II=I-1
0249                    ALS(II,1)=S(I,NX)
0250          170       CONTINUE
0251          175  CONTINUE
              CPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
              C
0252               IF(QUIT.EQ.0) GO TO 185
0253               IF(ITER.GT.QUIT) GO TO 500
0254          185  CONTINUE
              C
              C   CHECKS FOR OPTIMALITY,THAT IS PROPER VALUES OF A'S
              C   IF NOT OPTIMAL START AGAIN
0255               S(1,NX)=S(1,NX)-1.0D0C
0256                    DO 190 I=1,NP
0257                         IF(DABS(S(I,NX)).GT.ACC) GO TO 9876
0258          190            CONTINUE
              C
0259               GO TO 500
              C
0260          200  CONTINUE
0261               IF(TBV(NBVAT).LE.TC+DT) GO TO 55
0262               TC=TBV(NBVAT-1)
0263               GO TO 45
              C
              C-----------------------------------------------------------------------
              C-----------------------------------------------------------------------
              C
0264          500  CONTINUE
              C
0265               I=ITER-1
0266               IF(QUIT.EQ.0) GO TO 210
0267               IF(ITER-QUIT) 210,210,220
0268          210  WRITE(IPRT,900) I
0269          900  FORMAT(1H0,5X,10HITERATION ,I2,53H CONVERGED TO THE ACCURACY SPECI
                  1FIED FOR THE SOLUTION,2(/,125(1H*)))
0270               GO TO 230
0271          220  WRITE(IPRT,910) I
0272          910  FORMAT(1H0,5X,10HITERATION ,I2,82H HAS NOT CONVERGED TO THE ACCURA
                  1CY SPECIFIED IN THE NUMBER OF ALLOWABLE ITERATIONS,2(/,110(1H*)))
0273          230  CONTINUE
              C
              CPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
              C   IF THE INPUT VALUE OF STAT WAS 1,THEN A POST STAT PAK(PSTAT)
              C    WILL BE DONE ON THE RESULTS
              C
0274               IF(STAT.EQ.0) GO TO 510
0275               NDSTT=NDSTT+1
0276               IF(NDSTT.EQ.1) GO TO 9876
0277               NCW=NP-NEMAX
0278                  CALL PSTAT(NRZ,NRW,NCW,ALPHA,IOUT)
              C
              CPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
              C
```

```
                                                        MAIN5220
      510  CONTINUE                                     MAIN5230
           GO TO 1                                      MAIN5240
      520  CONTINUE                                     MAIN5250
           CALL EXIT                                    MAIN5260
           END
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = MAIN    , LINECNT =       60
*STATISTICS*    SOURCE STATEMENTS =      283,PROGRAM SIZE =    16108
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
      BLOCK DATA
C     BLOCK DATA ENABLES ALL DATA IN COMMON TO BE INITIALIZED
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8           YI(10),P(10,11),PC(20,10,11),TBV(25)
      REAL*8 W(25,10),ZHAT(25,1),ZEE(25,1),PB(10),YIR(10)
      REAL*8 ALS(10,1),DLS(25,1)
      COMMON/PYPS/YI,P,PC,TBV
      COMMON/REG/ZEE,ZHAT,W,PB,YIR,ALS,DLS,ZBAR
      DATA YI/10*0.0D0/
      DATA P/110*0.0D0/
      DATA PC/2200*0.0D0/
      DATA TBV/25*0.0D0/
      DATA ZEE/25*0.0D00/
      DATA ZHAT/25*0.0D00/
      DATA W/250*0.0D00/
      DATA YIR/10*0.0D0/
      DATA PB/10*0.0D0/
      DATA ALS/10*0.0D0/
      DATA DLS/25*0.0D0/
      END
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = BLK DATA, LINECNT =       60
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
0001         SUBROUTINE DTFIND(A,NEQ,NTERMS,RF,ACC,DI,DT)                   DTFND000
0002         IMPLICIT REAL*8(A-H,O-Z)                                       DTFND010
0003         REAL*8 A(20,20)                                                DTFND020
       C-----------------------------------------------------------------DTFND030
       C   DIFIND DETERMINES THE LARGEST STEP THAT CAN BE TAKEN             DTFND040
       C   IN INTEGRATING A POWER SERIES                                    DTFND050
       C   A IS THE POWER SERIES IN QUESTION                                DTFND060
       C   NEQ IS THE NUMBER OF NON-TRIVIAL DIFFERENTIAL EQ.                DTFND070
       C   NTERMS IS THE NUMBER OF TERMS IN THE POWER SERIES                DTFND080
       C   RF IS A FUDGE FACTOR, REDUCING THE STEP BY FACTOR OF RF          DTFND090
           ACC IS AN ACCURACY CHECK                                         DTFND100
       C   DI IS THE DESIRED STEP TO TAKE                                   DTFND110
       C   DT IS THE LARGEST STEP ALLOWED FOR THE GIVEN ACCURACY            DTFND120
       C-----------------------------------------------------------------DTFND130
0004         PMAXM1=NTERMS-2                                                DTFND140
0005         N=NTERMS-1                                                     DTFND150
0006         SMALL=1.0D-30                                                  DTFND160
0007         VS=1.0D06                                                      DTFND170
0008         DT=DABS(DI)                                                    DTFND180
0009                 DO 50 J=1,NEQ                                          DTFND190
       C                                                                    DTFND200
       C   THIS IF STATEMENT IS NEEDED TO AVOID DIVISION BY ZERO            DTFND210
       C    THIS CAN ARISE WHEN DEALING WITH UNKNOWN CONSTANTS              DTFND220
       C    THE REASON BEING THE FIRST TERM IS A CONSTANT,AND MAY BE ZERO,  DTFND230
       C    THE REMAINDER OF THE TERMS ARE ZERO IN THE POWER SERIES         DTFND240
       C                                                                    DTFND250
0010                       IF(A(N,J).EQ.0.0D0) GO TO 50                     DTFND260
       C                                                                    DTFND270
0011                 V=DABS(A(1,J)/A(N,J))                                  DTFND280
0012                 IF(V.LT.SMALL) V=SMALL                                 DTFND290
0013                 IF(VS.GT.V) VS=V                                       DTFND300
0014      50         CONTINUE                                               DTFND310
0015         DTT=DEXP(DLOG(ACC*VS)/PMAXM1)*RF                               DTFND320
0016         IF(DT.GT.DTT) DT=DTT                                           DTFND330
0017         DT=DT*DI/DABS(DI)                                              DTFND340
0018         RETURN                                                         DTFND350
0019         END                                                            DTFND360
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = DTFIND  , LINECNT =       60
*STATISTICS*    SOURCE STATEMENTS =       19,PROGRAM SIZE =       890
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
FORTRAN IV G LEVEL  21               DEVAL          DATE = 76097        15/49/10            PAGE 0001

0001          SUBROUTINE DEVAL(A,SUM,NEQ,NTERMS,ACC,DT)                    DEVAL000
0002          IMPLICIT REAL*8(A-H,O-Z)                                     DEVAL010
0003          REAL*8 A(20,20),SUM(20)                                      DEVAL020
0004          COMMON/LOG/INPT,IPRT                                         DEVAL030
        C------------------------------------------------------------------DEVAL040
        C   DEVAL EVALUATES THE DERIVATIVE OF A POWER SERIES               DEVAL050
        C   A IS THE VECTOR OF TERMS OF WHICH THE DERIVATIVE IS DESIRED    DEVAL060
        C   SUM IS THE VALUE OF THE DERIVATIVE                             DEVAL070
        C   NEQ IS THE NUMBER OF NON-TRIVIAL DIFFERENTIAL EQ.              DEVAL080
        C   NTERMS IS THE NUMBER OF TERMS IN THE POWER SERIES              DEVAL090
        C   ACC IS AN ACCURACY CHECK                                       DEVAL100
        C   DT IS THE TIME INCREMENT FOR INTEGRATION FROM CENTER OF EXPANSION  DEVAL110
        C------------------------------------------------------------------DEVAL120
0005          DO 10 J=1,NEQ                                                DEVAL130
0006          TPOW=1.D00                                                   DEVAL140
0007          PSA=A(2,J)                                                   DEVAL150
0008          ICHK=0                                                       DEVAL160
0009          DO 5 I=3,NTERMS                                              DEVAL170
0010          FI=DFLOAT(I-1)                                               DEVAL180
0011          TPOW=TPOW*DT                                                 DEVAL190
0012          TT=A(I,J)*TPOW*FI                                            DEVAL200
0013          PSA=PSA+TT                                                   DEVAL210
        C                                                                  DEVAL220
        C   THIS IF STATEMENT IS NEEDED TO AVOID DIVISION BY ZERO          DEVAL230
        C    WHERE THERE MAY BE ONLY ONE NON-ZERO TERM IN THE POWER SERIES DEVAL250
        C    THIS CAN ARISE WHEN DEALING WITH UNKNOWN CONSTANTS            DEVAL240
        C                                                                  DEVAL260
0014          IF(PSA.EQ.0.0D0) PSA=1.0D-78                                 DEVAL270
        C                                                                  DEVAL280
0015          IF(DABS(TT/PSA).LT.ACC) ICHK=ICHK+1                          DEVAL290
0016          IF(ICHK.EQ.2) GO TO 10                                       DEVAL300
0017        5 CONTINUE                                                     DEVAL310
0018          WRITE(IPRT,900) NEQ,J,ICHK,TT                                DEVAL320
0019      900 FORMAT(1X,10HNEQ,J,ICHK,3I4,3X,4HTT =,G15.8,/)               DEVAL330
0020       10 SUM(J)=PSA                                                   DEVAL340
0021          RETURN                                                       DEVAL350
0022          END                                                          DEVAL360
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = DEVAL    , LINECNT =       60
*STATISTICS*    SOURCE STATEMENTS =       22,PROGRAM SIZE =      864
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
FORTRAN IV G LEVEL  21                EVAL               DATE = 76097        15/49/10            PAGE 0001

0001              SUBROUTINE EVAL(A,SUM,NEQ,NTERMS,ACC,DT)                                 EVAL0000
0002              IMPLICIT REAL*8(A-H,O-Z)                                                 EVAL0010
0003              REAL*8 A(20,20),SUM(20)                                                  EVAL0020
0004              COMMON/LOG/INPT,IPRT                                                     EVAL0030
           C-----------------------------------------------------------------------EVAL0040
           C    EVAL EVALUATES A POWER SERIES                                              EVAL0050
           C    A IS THE POWER SERIES TO BE INTEGRATED                                     EVAL0060
           C    SUM IS THE INTEGRATED VALUE OF THE POWER SERIES                            EVAL0070
           C    NEQ IS THE NUMBER OF NON-TRIVIAL DIFFERENTIAL EQ.                          EVAL0080
           C    NTERMS IS THE NUMBER OF TERMS IN THE POWER SERIES                          EVAL0090
           C    ACC IS AN ACCURACY CHECK                                                   EVAL0100
           C    DT IS THE TIME INCREMENT FOR INTEGRATION FROM CENTER OF EXPANSION          EVAL0110
           C-----------------------------------------------------------------------EVAL0120
0005              DO 10 J=1,NEQ                                                            EVAL0130
0006              TPOW=1.D00                                                               EVAL0140
0007              PSA=A(1,J)                                                               EVAL0150
0008              ICHK=0                                                                   EVAL0160
0009              DO 5 I=2,NTERMS                                                          EVAL0170
0010              TPOW=TPOW*DT                                                             EVAL0180
0011              TT=A(I,J)*TPOW                                                           EVAL0190
0012              PSA=PSA+TT                                                               EVAL0200
           C                                                                               EVAL0210
           C    THIS IF STATEMENT IS NEEDED TO AVOID DIVISION BY ZERO                      EVAL0220
           C     THIS CAN ARISE WHEN DEALING WITH UNKNOWN CONSTANTS                        EVAL0230
           C     WHERE THERE MAY BE ONLY ONE NON-ZERO TERM IN THE POWER SERIES             EVAL0240
           C                                                                               EVAL0250
0013              IF(PSA.EQ.0.0L0) PSA=1.0D-78                                             EVAL0260
           C                                                                               EVAL0270
0014              IF(DABS(TT/PSA).LT.ACC) ICHK=ICHK+1                                      EVAL0280
0015              IF(ICHK.EQ.2) GO TO 10                                                   EVAL0290
0016            5 CONTINUE                                                                 EVAL0300
0017              WRITE(IPRT,900) NEQ,J,ICHK,TT                                            EVAL0310
0018        900   FORMAT(1X,10HNEQ,J,ICHK,3I4,3X,4HTT =,G15.8,/)                           EVAL0320
0019           10 SUM(J)=PSA                                                               EVAL0330
0020              RETURN                                                                   EVAL0340
0021              END                                                                      EVAL0350
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = EVAL    , LINECNT =       60
*STATISTICS*    SOURCE STATEMENTS =       21,PROGRAM SIZE =      804
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
0001         SUBROUTINE GJRWLS                                                  GJLS0000
            1(A,B,NRA,NRB,NEMAX,DET,CHECK,IOUT,N,NROWA,IPUNT,IGRAD,STAT,W,DLS) GJLS0010
0002         IMPLICIT REAL*8(A-H,O-Z)                                           GJLS0020
0003         REAL*8 A(20,20),B(20,20),W(25,10)                                  GJLS0030
0004         REAL*8 DLS(25,1)                                                   GJLS0040
0005         INTEGER I(101),J(52),STAT                                          GJLS0050
0006         COMMON/LOG/INPT,IPRT                                               GJLS0060
       C ---------------------------------------------------------------------- GJLS0070
       C THIS ROUTINE SOLVES THE SYSTEM OF SIMULTANEOUS EQUATIONS               GJLS0080
       C IF THE SYSTEM IS OVERDETERMINED IT CALLS A ROUTINE TO REDUCE           GJLS0090
       C THE NONSQUARE MATRIX TO A SQUARE ONE                                   GJLS0100
       C THE METHOD USED IS GAUSS JORDAN WITH PARTIAL PIVOTING                  GJLS0110
       C----------------------------------------------------------------------- GJLS0120
0007            NTOTAL = N + 1                                                  GJLS0130
0008         LSTOP=NEMAX                                                        GJLS0140
0009         NROW=NROWA                                                         GJLS0150
0010         NEND=NROWA                                                         GJLS0160
0011         IF(NROWA.LT.NTOTAL) NEND=NTOTAL                                    GJLS0170
0012                 DO 100 L=1,NEND                                            GJLS0180
0013         IF(L.LE.NTOTAL) J(L)=L                                             GJLS0190
0014     100 I(L)=L                                                             GJLS0200
0015            IF(NROWA.NE.N.AND.NEMAX.EQ.0)CALL  LSTSQS                       GJLS0210
            1( A, B, NRA, NRB, NEMAX, J, LSTOP, NROW, N, NROWA, IGRAD)          GJLS0220
0016         DO 190 K=1,N                                                       GJLS0230
0017         C=0.0D00                                                           GJLS0240
0018         DO 120 L=K,LSTOP                                                   GJLS0250
0019         IL = I(L)                                                          GJLS0260
0020                    DO 120 M= K,N                                           GJLS0270
0021         JM=J(M)                                                            GJLS0280
0022         ABSA=DABS(A(IL,JM))                                                GJLS0290
0023         IF (ABSA.LE.C)  GO TO 120                                          GJLS0300
0024         KL = L                                                             GJLS0310
0025         KM = M                                                             GJLS0320
0026         C = ABSA                                                           GJLS0330
0027     120 CONTINUE                                                           GJLS0340
0028         KI = I(KL)                                                         GJLS0350
0029         KJ = J(KM)                                                         GJLS0360
0030                    I(KL) = I(K)                                            GJLS0370
0031         I(K) = KI                                                          GJLS0380
0032         J(KM) = J(K)                                                       GJLS0390
0033         J(K) = KJ                                                          GJLS0400
0034         AKK = A(KI,KJ)                                                     GJLS0410
0035         A(KI,KJ)=1.D00                                                     GJLS0420
0036         DET = DET*AKK                                                      GJLS0430
0037         IF (NEMAX.GT.N .AND. K.GE.N) GO TO 260                             GJLS0440
0038         IF(C.GT.CHECK) GO TO 140                                           GJLS0450
0039         IPUNT = 2                                                          GJLS0460
0040         RETURN                                                             GJLS0470
       C                                                                        GJLS0480
       CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDGJLS0490
0041      140  IF(IOUT.GT.1) WRITE(IPRT,900) K,AKK,KI,KJ                        GJLS0500
0042      900  FORMAT(20X,I5,9HTH PIVOT ,E15.7,2I5)                             GJLS0510
       CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDGJLS0520
       C                                                                        GJLS0530
0043         AKK = 1.D0/AKK                                                     GJLS0540
0044         KP1=K+1                                                            GJLS0550
0045         DO 160 M=KP1,NTOTAL                                                GJLS0560
0046         JM=J(M)                                                            GJLS0570
```

```
0047        160 A(KI,JM)=AKK*A(KI,JM)                                        GJLS0580
0048            DO 180 L=1,NROW                                              GJLS0590
0049            IF(L.EQ.K) GO TO 180                                         GJLS0600
0050                             IL = I(L)                                   GJLS0610
0051            AKK = A(IL,KJ)                                               GJLS0620
0052            A(IL,KJ)=0.0D0                                               GJLS0630
0053            DO 170 M=KP1,NTOTAL                                          GJLS0640
0054            JM = J(M)                                                    GJLS0650
0055        170 A(IL,JM) = A(IL,JM) - AKK*A(KI,JM)                           GJLS0660
0056        180 CONTINUE                                                     GJLS0670
0057            IF (K.NE.NEMAX .OR. N.LE.K) GO TO 190                        GJLS0680
          C                                                                  GJLS0690
          CPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPGJLS0700
0058            IF(STAT.EQ.0) GO TO 195                                      GJLS0710
0059            NSTART=NEMAX+1                                               GJLS0720
0060            DO 185 II=NSTART,NRA                                         GJLS0730
0061            III=II-NEMAX                                                 GJLS0740
0062            DO 185 JJ=NSTART,N                                           GJLS0750
0063            JJJ=JJ-NEMAX                                                 GJLS0760
0064            W(III,JJJ)=A(II,JJ)                                          GJLS0770
0065            DLS(III,1)=A(II,NTOTAL)                                      GJLS0780
0066       185  CONTINUE                                                     GJLS0790
0067       195  CONTINUE                                                     GJLS0800
          CPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPGJLS0810
          C                                                                  GJLS0820
          CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDGJLS0830
0068            IF(IOUT.NE.-1) GO TO 10                                      GJLS0840
0069                  DO 5 JP=1,20                                           GJLS0850
0070       5          WRITE(IPRT,910)(A(JP,KP),KP=1,10)                      GJLS0860
0071       910  FORMAT(1H0,10(G12.5))                                        GJLS0870
0072       10   CONTINUE                                                     GJLS0880
          CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDGJLS0890
          C                                                                  GJLS0900
0073                              CALL   LSTSQS                              GJLS0910
             1( A, B, NRA, NRB,   K,    J, LSTOP, NROW, N, NROWA, IGRAD)     GJLS0920
0074        190 CONTINUE                                                     GJLS0930
0075                          DO 240 K=1,N                                   GJLS0940
0076            KI = I(K)                                                    GJLS0950
0077            KJ = J(K)                                                    GJLS0960
0078            IF(KI.EQ.KJ) GO TO 210                                       GJLS0970
0079            DO 200 M=1,NTOTAL                                            GJLS0980
0080            AKK=A(KI,M)                                                  GJLS0990
0081            A(KI,M) = A(KJ,M)                                            GJLS1000
0082        200 A(KJ,M) = AKK                                                GJLS1010
0083        210 DO 220 L = K,N                                               GJLS1020
0084            IF(I(L).EQ.J(K)) GO TO 230                                   GJLS1030
0085        220 CONTINUE                                                     GJLS1040
0086        230 I(L) = I(K)                                                  GJLS1050
0087        240 I(K) = L                                                     GJLS1060
          C                                                                  GJLS1070
          CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDGJLS1080
0088       260  IF(IOUT.GT.0) WRITE(IPRT,920) DET                            GJLS1090
0089       920  FORMAT(10X,6HDET = ,E15.7)                                   GJLS1100
          CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDGJLS1110
          C                                                                  GJLS1120
0090            IPUNT=1                                                      GJLS1130
0091            RETURN                                                       GJLS1140
0092            END                                                          GJLS1150
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = GJRWLS  , LINECNT =       60
*STATISTICS*    SOURCE STATEMENTS =       92,PROGRAM SIZE =      3600
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
FORTRAN IV G LEVEL  21              LSTSQS           DATE = 76097        15/49/10              PAGE 0001

0001           SUBROUTINE LSTSQS(A,B,NRA,NRB,K,J,LSTOP,NROW,N,NROWA,IGRAD)           LSTSQ000
0002           IMPLICIT REAL*8(A-H,O-Z)                                              LSTSQ010
0003           REAL*8 A(20,20),B(20,20)                                              LSTSQ020
         C     REAL      A(NRA,1),  B(NRB,1),  HBIG(  270)                           LSTSQ030
0004           INTEGER JZC(20)                                                       LSTSQ040
0005           INTEGER J(1)                                                          LSTSQ050
         C-----------------------------------------------------------------          LSTSQ060
         C THIS ROUTINE WHEN THE SAVE MATRIX IS OVERDETERMINED                        LSTSQ070
         C FORMS THE SQUARE PRODUCT OF THE SAVE MATRIX AND ITS                        LSTSQ080
         C TRANSPOSE                                                                  LSTSQ090
         C-----------------------------------------------------------------          LSTSQ100
0006           KP1=K+1                                                               LSTSQ110
0007           NP1=N+1                                                               LSTSQ120
0008           DO 100 L=1,N                                                          LSTSQ130
0009       100           JZC(L) = 0                                                  LSTSQ140
0010                                                        BTB=0.D0                 LSTSQ150
0011                                                        DO 110  M=KP1,NROWA      LSTSQ160
0012       110                                            BTB=BTB+A(M,NP1)**2        LSTSQ170
0013                                                                 IRA = 0         LSTSQ180
0014           DO 140 L=KP1,N                                                        LSTSQ190
0015                                                                 IRA=IRA + 1     LSTSQ200
0016                                                                 IRB = 0         LSTSQ210
0017           IRAT=J(L)                                                             LSTSQ220
0018           JZC(IRAT)=1                                                           LSTSQ230
0019           DO 130 M=KP1,NP1                                                      LSTSQ240
0020           JCA=J(M)                                                              LSTSQ250
0021                                                                 IRB=IRB + 1     LSTSQ260
0022           IF(IRAT.GT.JCA) GO TO 130                                             LSTSQ270
0023           TEMP=0.D00                                                            LSTSQ280
0024           DO 120 KR=KP1,NROWA                                                   LSTSQ290
0025       120 TEMP=TEMP+A(KR,IRAT)*A(KR,JCA)                                        LSTSQ300
0026                                                        B(IRA,IRB) = TEMP        LSTSQ310
0027                         IF(IRAT.LT.JCA.AND.JCA.LE.N) B(IRB,IRA) = TEMP          LSTSQ320
0028       130 CONTINUE                                                              LSTSQ330
0029       140 CONTINUE                                                              LSTSQ340
0030                                         IRB = 0                                 LSTSQ350
0031                     DO 160 L = KP1, N                                           LSTSQ360
0032                                         IRB = IRB + 1                           LSTSQ370
0033                                         JCB = 0                                 LSTSQ380
0034           DO 150 M=KP1,NP1                                                      LSTSQ390
0035           JCA=J(M)                                                              LSTSQ400
0036                                         JCB = JCB + 1                           LSTSQ410
0037       150                                          A(L,JCA) = B(IRB,JCB)        LSTSQ420
0038       160 CONTINUE                                                              LSTSQ430
0039           LSTOP=N                                                               LSTSQ440
0040           NROW=N                                                                LSTSQ450
0041                     RETURN                                                      LSTSQ460
0042                         END                                                     LSTSQ470
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = LSTSQS  , LINECNT =       60
*STATISTICS*    SOURCE STATEMENTS =        42,PROGRAM SIZE =      1506
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
FORTRAN IV G LEVEL  21              CALCSR           DATE = 76097          15/49/10              PAGE 0001

0001           SUBROUTINE CALCSR(IOUT)                                                 CALCS000
0002           IMPLICIT REAL*8(A-H,O-Z)                                                CALCS010
0003           REAL*8 YI(10),P(10,11),PC(20,10,11),TBV(25)                             CALCS020
0004           COMMON/PYPS/YI,P,PC,TBV                                                 CALCS030
0005           COMMON/PARM/TO,TC,NDE,NP,NBV,NTERMS,LINEAR                              CALCS040
0006           COMMON/LOG/INPT,IPRT                                                    CALCS050
         C-----------------------------------------------------------------------------CALCS060
         C   CALCSR CALCULATES THE POWER SERIES ELEMENTS NEEDED                        CALCS070
         C   RECURL IS CALLED FROM THIS SUBPROGRAM                                     CALCS080
         C-----------------------------------------------------------------------------CALCS090
0007                 DO 100 I=1,NP                                                     CALCS100
0008                 CALL RECURL(P(1,I),PC(1,1,I),I)                                   CALCS110
         C                                                                             CALCS120
         CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDCALCS130
0009                 IF(IOUT.NE.4) GO TO 100                                           CALCS140
0010                 WRITE(IPRT,900)I                                                  CALCS150
0011      900        FORMAT(1H1,11HFROM CALCSR,3X,I5,3X,19HPARTICULAR SOL COEF)CALCS160
0012                 DO 25 J=1,NTERMS                                                  CALCS170
0013                 WRITE(IPRT,910)(PC(J,K,I),K=1,NDE)                                CALCS180
0014      25         CONTINUE                                                          CALCS190
0015      910        FORMAT(1H ,5X,10G12.5)                                            CALCS200
         CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDCALCS210
         C                                                                             CALCS220
0016      100        CONTINUE                                                          CALCS230
0017           RETURN                                                                  CALCS240
0018           END                                                                     CALCS250
```

```
FORTRAN IV G LEVEL  21                CALCSR            DATE = 76097          15/49/10              PAGE 0002

  *OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
  *OPTIONS IN EFFECT*  NAME = CALCSR   , LINECNT =       60
  *STATISTICS*    SOURCE STATEMENTS =        18,PROGRAM SIZE =       672
  *STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
CRTRAN IV G LEVEL  21                    RECURL              DATE = 76097           15/49/10                 PAGE 0001

0001            SUBROUTINE RECURL(Y,YC,IND)                                          RECRL000
0002            IMPLICIT REAL*8(A-H,O-Z)                                             RECRL010
0003            REAL*8 YI(10),P(10,11),PC(20,10,11),TBV(25)                          RECRL020
0004            REAL*8 SN(20),CS(20),TT(20)                                          RECRL030
0005            REAL*8 Y(10),YC(20,10)                                               RECRL040
0006            REAL*8 TEMP2(20)                                                     RECRL050
0007            COMMON/PARM/TO,TC,NDE,NP,NBV,NTERMS,LINEAR                           RECRL060
0008            COMMON/PYPS/YI,P,PC,TBV                                              RECRL070
          C----------------------------------------------------------------------RECRL080
          C   RECURL GOES THROUGH THE RECURSIVE RELATIONSHIPS FOR THE               RECRL090
          C   POWER SERIES AND SENDS IT BACK TO CALCSR                              RECRL100
          C   Y IS AN INPUT MATRIX                                                  RECRL110
          C   YC IS THE RESULTING MATRIX                                            RECRL120
          C   IND INDICATES FOR WHICH PARTICULAR SOLUTION                           RECRL130
          C   FOR MASS-SPRING-DAMPER SYSTEM                                         RECRL140
          C----------------------------------------------------------------------RECRL150
0009            NTIMES=NTERMS-1                                                      RECRL160
0010            IF(IND.GT.1) GO TO 50                                                RECRL170
0011               DO 10 I=1,20                                                      RECRL180
0012               TT(I)=0.0D0                                                       RECRL190
0013       10      CONTINUE                                                          RECRL200
0014            TT(1)=TC                                                             RECRL210
0015            TT(2)=1.0D0                                                          RECRL220
0016            CALL TRIGF(TT,SN,CS,20,0,1)                                          RECRL230
0017       50      DO 60 I=1,NDE                                                     RECRL240
0018               YC(1,I)=Y(I)                                                      RECRL250
0019       60      CONTINUE                                                          RECRL260
          C----------------------------------------------------------------------RECRL270
          C   DO 100 LOOP CHANGES FOR GVEN PROGRAM                                  RECRL280
          C   THEREFORE MUST BE REPROGRAMED                                         RECRL290
          C                                                                         RECRL300
0020            DO 100 I=1,NTIMES                                                    RECRL310
0021            FI=DFLOAT(I)                                                         RECRL320
0022            YC(I+1,1)=YC(I,2)/FI                                                 RECRL330
0023            IF(IND.GT.1.AND.LINEAR.EQ.0) GO TO 70                                RECRL340
0024            YC(I+1,2)=(SN(I)-Y(4)*YC(I,1)-Y(3)*YC(I,2))/FI                       RECRL350
0025            TEMP2(I)=SN(I)+YC(I,1)*Y(4)+YC(I,2)*Y(3)                             RECRL360
0026            GO TO 100                                                            RECRL370
0027       70   CONTINUE                                                             RECRL380
0028            TEMP1=0.0D0-P(4,1)*YC(I,1)-P(3,1)*YC(I,2)-PC(I,2,1)*Y(3)             RECRL390
               A-PC(I,1,1)*Y(4)                                                      RECRL400
0029            YC(I+1,2)=(TEMP1+TEMP2(I))/FI                                        RECRL410
0030      100   CONTINUE                                                             RECRL420
          C----------------------------------------------------------------------RECRL430
0031            RETURN                                                               RECRL440
0032            END                                                                  RECRL450
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = RECURL  , LINECNT =       60
*STATISTICS*    SOURCE STATEMENTS =       32,PROGRAM SIZE =     1522
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
FORTRAN IV G LEVEL  21              TRIGF              DATE = 76097          15/49/10              PAGE 0001

0001            SUBROUTINE TRIGF(TH,SN,CS,INDX,IFLAG,ISTRT)                         TRIGF000
0002            IMPLICIT REAL*8(A-H,O-Z)                                            TRIGF010
0003            REAL*8 TH(20),SN(20),CS(20)                                         TRIGF020
          C---------------------------------------------------------------------TRIGF030
          C   TH ARE VALUES OF THE COEFFICIENTS OF THE POWER SERIES             TRIGF040
          C    FOR EITHER THETA OR THETA-DOT                                    TRIGF050
          C   SN IS THE VALUE OF THE FIRST COEF IN SINE POWER SERIES SIN(THETA(O))TRIGF060
          C   CS IS THE VALUE OFTHE FIRST COEF IN COSINE PWER SERIES C/S(THETA(O))TRIGF070
          C   INDX IS THE INDEX NUMBER OF THE POWER S RIES AND IS I+1           TRIGF080
          C   IFLAG INDICATES WHETHER INPUT IS THETA OR THETA DOT               TRIGF090
          C    IFLAG=0    IMPLIES THETA                                         TRIGF100
          C    IFLAG=1    IMPLIES THETA DOT                                     TRIGF110
          C   ISTRT IS THE POINT IN THE SERIES AT WHICH YOU DESIRE TO START     TRIGF120
          C   MINIMUM VALUE FOR ISTRT IS 1                                      TRIGF130
          C---------------------------------------------------------------------TRIGF140
0004            IF (IFLAG.EQ.1) GO TO 20                                            TRIGF150
0005            IF(ISTRT.GT.1) GO TO 5                                              TRIGF160
0006            CS(1)=DCOS(TH(1))                                                   TRIGF170
0007            SN(1)=DSIN(TH(1))                                                   TRIGF180
0008            IF(INDX.LE.1)RETURN                                                 TRIGF190
0009       5    JSTRT=2                                                             TRIGF200
0010            IF(ISTRT.GT.JSTRT) JSTRT=ISTRT                                      TRIGF210
          C                                                                     TRIGF220
          C     THETA INPUT CALCULATIONS                                        TRIGF230
0011                    DO 15 J=JSTRT,INDX                                          TRIGF240
0012                    SN(J)=0.0D0                                                 TRIGF250
0013                    CS(J)=0.0D0                                                 TRIGF260
0014                    I=J-1                                                       TRIGF270
0015                         DO 10 KJ=1,I                                           TRIGF280
0016                         JSS=J-KJ                                               TRIGF290
0017                         CS(J)=CS(J)+DFLOAT(KJ)*SN(JSS)*TH(KJ+1)                TRIGF300
0018      10                 SN(J)=SN(J)+DFLOAT(KJ)*CS(JSS)*TH(KJ+1)                TRIGF310
0019                    CS(J)=-CS(J)/DFLOAT(J-1)                                    TRIGF320
0020      15            SN(J)=SN(J)/DFLOAT(J-1)                                     TRIGF330
0021            GO TO 35                                                            TRIGF340
          C                                                                     TRIGF350
          C     THETA DOT INPUT CALCULATIONS                                    TRIGF360
0022      20    CONTINUE                                                            TRIGF370
0023                    DO 30 J=ISTRT,INDX                                          TRIGF380
0024                    SN(J)=0.0D0                                                 TRIGF390
0025                    CS(J)=0.0D0                                                 TRIGF400
0026                    I=J-1                                                       TRIGF410
0027                         DO 25 KJ=1,I                                           TRIGF420
0028                         JSS=J-KJ                                               TRIGF430
0029                         CS(J)=CS(J)+SN(JSS)*TH(KJ)                             TRIGF440
0030      25                 SN(J)=SN(J)+CS(JSS)*TH(KJ)                             TRIGF450
0031                    CS(J)=-CS(J)/DFLOAT(J-1)                                    TRIGF460
0032      30            SN(J)=SN(J)/DFLOAT(J-1)                                     TRIGF470
          C                                                                     TRIGF480
0033      35    CONTINUE                                                            TRIGF490
0034            RETURN                                                              TRIGF500
0035            END                                                                 TRIGF510
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = TRIGF    , LINECNT =       60
*STATISTICS*     SOURCE STATEMENTS =        35,PROGRAM SIZE =      1286
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
FORTRAN IV G LEVEL  21              PSTAT              DATE = 76097           15/49/10              PAGE 0001

0001           SUBROUTINE PSTAT(NRZ,NRW,NCW,ALPHA,IOUT)                            PSTT0000
0002           IMPLICIT REAL*8(A-H,C-Z)                                            PSTT0010
0003           REAL*4 SNCW,SIDF,SALPHA,SPROB,SFVAL,STVAL,SFCAL                     PSTT0020
0004           REAL*8 W(25,10),ZHAT(25,1),ZEE(25,1),PB(10),YIR(10)                 PSTT0030
0005           REAL*8 ALS(10,1),DLS(25,1),BCLL(25),BCUL(25),R3(1,10),RSDUL(25)     PSTT0040
0006           REAL*8 WK(1,10),WKT(10,1),VARD(1,1),SEBC(25)                        PSTT0050
0007           REAL*8 LIMYI(10),WORK(10),R2I(10,10),VARYI(10)                      PSTT0060
0008           REAL*8 WT(10,25),REG(1,1),TOT(1,1),R2(10,10),VARB(10,10)            PSTT0070
0009           REAL*8 R1(1,25),REG2(1,1),TOT2(1,1)                                 PSTT0080
0010           REAL*8 MSRES,MSREG,MS2                                              PSTT0090
0011           COMMON/PARM/TC,TC,NDE,NP,NBV,NTERMS,LINEAR                          PSTT0100
0012           COMMON/REG/ZEE,ZHAT,W,PB,YIR,ALS,DLS,ZBAR                           PSTT0110
0013           COMMON/LOG/INPT,IPRT                                                PSTT0120
0014           DATA WT/250*0.0D0/                                                  PSTT0130
0015           DATA WORK/10*0.0D0/                                                 PSTT0140
0016           DATA VARB/100*0.0D0/                                                PSTT0150
0017           DATA VARYI/10*0.0D0/                                                PSTT0160
0018           DATA SEBC/25*0.0D0/                                                 PSTT0170
0019           DATA LIMYI/10*0.0D0/                                                PSTT0180
0020           DATA WK/10*0.0D0/                                                   PSTT0190
0021           DATA WKT/10*0.0D0/                                                  PSTT0200
0022           DATA RSDUL/25*0.0D0/                                                PSTT0210
0023           DATA SMRES/0.0D0/                                                   PSTT0220
0024           DATA R1/25*0.0D0/                                                   PSTT0230
0025           DATA R2/100*0.0D0/                                                  PSTT0240
0026           DATA R2I/100*0.0D0/                                                 PSTT0250
0027           DATA R3/10*0.0D0/                                                   PSTT0260
         C---------------------------------------------------------------------PSTT0270
         C   PSTAT IS A STATISTICAL ANALYSIS WHICH IS USED ONLY IF STAT=1       PSTT0280
         C    THE METHOD OF SOLUTION IS FROM DRAPER & SMITH,1966                PSTT0290
         C    AND THE FORM OF THE REGRSSION IS Z=WB + ERROR                     PSTT0300
         C   VARIABLE DEFINITION                                                PSTT0310
         C    NRZ IS NUMBER OF ROWS OF ZEE AND ZHAT MATRICES                    PSTT0320
         C    NRW IS THE NUMBER OF ROWS OF THE W MATRIX                         PSTT0330
         C    NCW IS THE NUMBER OF COLUMNS OF THE W MATRIX                      PSTT0340
         C   ALPHA IS THE CONSUMERS RISK                                        PSTT0350
         C   IOUT IS A DEBUG VARIABLE. ONLY APPLIES TO PSTAT WHEN =-1           PSTT0360
         C                                                                      PSTT0370
         CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDPSTT0380
0028           IF(IOUT.NE.-1) GO TO 20                                             PSTT0390
0029           WRITE(IPRT,600)NRZ,NRW,NCW,ALPHA                                    PSTT0400
0030       600 FORMAT(1H1,5X,16HFROM PSTAT-DEBUG,/,10X,                            PSTT0410
              A4HNRZ=,I5,2X,4HNRW=,I5,2X,4HNCW=,I5,2X,6HALPHA=,G15.8,//)           PSTT0420
0031           DO 10 I=1,25                                                        PSTT0430
0032           WRITE(IPRT,610)(W(I,J),J=1,10)                                      PSTT0440
0033        10 CONTINUE                                                            PSTT0450
0034       610 FORMAT(1H ,1HW,4X,10(G12.5),/)                                      PSTT0460
0035           WRITE(IPRT,620)(ZEE(I,1),I=1,25)                                    PSTT0470
0036       620 FORMAT(1H0,3HZEE,2X,G15.8)                                          PSTT0480
0037           WRITE(IPRT,630)(ZHAT(I,1),I=1,25)                                   PSTT0490
0038       630 FORMAT(1H0,4HZHAT,1X,G15.8)                                         PSTT0500
0039           WRITE(IPRT,640)(PB(I),I=1,10)                                       PSTT0510
0040       640 FORMAT(1H0,2HPB,3X,10(G12.5),//)                                    PSTT0520
0041           WRITE(IPRT,650)(YIR(I),I=1,10)                                      PSTT0530
0042       650 FORMAT(1H0,3HYIR,2X,10(G12.5))                                      PSTT0540
0043           WRITE(IPRT,660)(ALS(I,1),I=1,10)                                    PSTT0550
0044       660 FORMAT(1H0,3HALS,2X,10(G12.5))                                      PSTT0560
```

```
0046      670  FORMAT(1H0,3HCLS,2X,G15.8)                                       PSTT0580
0047       20  CONTINUE                                                          PSTT0590
         CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDPSTT0600
         C                                                                       PSTT0610
         C-----------------------------------------------------------------------PSTT0620
         C    THIS FIRST SECTION IS THE MAJOR MATRIX OPERATIONS NEEDED            PSTT0630
         C    THE FOLLOWING ARE A RESULT OF THIS SECTION                          PSTT0640
         C     REG  SUM OF SQUARES DUE TO REGRESSION                              PSTT0650
         C     RES  THE RESIDUAL OR THE SUM OF SQUARES ABOUT THE REGRESSION       PSTT0660
         C     TOT  THE TOT SUM OF SQUARES UNCORRECTED FOR THE MEAN               PSTT0670
         C                                                                       PSTT0680
0048              CALL DTPRD(ZHAT,ZEE,REG,25,1,0,0,1)                            PSTT0690
0049              CALL DMATA(ZEE,TOT,25,1,0)                                     PSTT0700
0050           RES=TOT(1,1)-REG(1,1)                                             PSTT0710
         C                                                                       PSTT0720
         CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDPSTT0730
0051           IF(IOUT.NE.-1) GO TO 30                                           PSTT0740
0052           WRITE(IPRT,680)REG(1,1)                                           PSTT0750
0053      680  FORMAT(1H0,3HREG,2X,G15.8,//)                                     PSTT0760
0054           WRITE(IPRT,690) RES                                               PSTT0770
0055      690  FORMAT(1H0,3HRES,2X,G15.8,/)                                      PSTT0780
0056           WRITE(IPRT,700)TOT(1,1)                                           PSTT0790
0057      700  FORMAT(1H0,3HTOT,2X,G15.8,//)                                     PSTT0800
0058       30  CONTINUE                                                          PSTT0810
         CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDPSTT0820
         C                                                                       PSTT0830
         C-----------------------------------------------------------------------PSTT0840
         C    CALCULATE THE MEAN SQUARE TERMS                                     PSTT0850
0059           IDF=NRZ-NCW                                                       PSTT0860
0060           MSREG=REG(1,1)/DFLOAT(NCW)                                        PSTT0870
0061           MSRES=DABS(RES/DFLOAT(IDF))                                       PSTT0880
         C                                                                       PSTT0890
         C    CALCULATE THE CORRELATION FACTOR, F CALCULATED, AND THE             PSTT0900
         C     COEFICIENT OF VARIATION                                            PSTT0910
0062           ZCOR=NRZ*(ZBAR**2)                                                PSTT0920
0063           RSQ=((REG(1,1)-ZCOR)/(TOT(1,1)-ZCOR))*100.000                     PSTT0930
0064           CV=DABS(DSQRT(MSRES)/ZBAR)                                        PSTT0940
0065           FCAL=MSREG/MSRES                                                  PSTT0950
         C                                                                       PSTT0960
         C-----------------------------------------------------------------------PSTT0970
         C    DETERMINE F AND T VALUES FOR GIVEN ALPHA                            PSTT0980
         C    THESE SUBROUTINES ARE FROM THE IMSL LIBRARY 1                       PSTT0990
0066           SNCW=FLOAT(NCW)                                                   PSTT1000
0067           SIDF=FLOAT(IDF)                                                   PSTT1010
0068           SALPHA=ALPHA                                                      PSTT1020
0069           SPROB=1.000-ALPHA                                                 PSTT1030
0070           SFCAL=FCAL                                                        PSTT1040
0071              CALL MDFI(SPROB,SNCW,SIDF,SFVAL,IER)                           PSTT1050
0072              CALL MDSTI(ALPHA,SIDF,STVAL,IER)                               PSTT1060
0073              CALL MDFD(SFCAL,NCW,IDF,SPROB,IER)                             PSTT1070
0074           FVAL=SFVAL                                                        PSTT1080
0075           TVAL=STVAL                                                        PSTT1090
0076           PROB=SPROB                                                        PSTT1100
0077           PROB=1.000-PROB                                                   PSTT1110
         C                                                                       PSTT1120
         C-----------------------------------------------------------------------PSTT1130
         C    THRU TO LOOP 100 CALCULATES THE COVARIANCE MATRIX                   PSTT1140
         C     OF THE SUPERPOSITION CONSTANTS                                     PSTT1150
```

```
          C     THE VARIANCE TERMS WILL BE ON THE DIAGONAL OF THE MATRIX        PSTT1160
          C     THAT IS, VARB(I,I)                                              PSTT1170
          C                                                                     PSTT1180
 0078                CALL DMTRA(W,WT,25,10,0)                                   PSTT1190
 0079                CALL DMPRD(WT,W,R2,10,25,0,0,10)                           PSTT1200
 0080                CALL DTPRD(ALS,WT,R1,10,1,0,0,25)                          PSTT1210
 0081                CALL DMPRD(R1,DLS,REG2,1,25,0,0,1)                         PSTT1220
 0082                CALL DMATA(DLS,TOT2,25,1,0)                                PSTT1230
 0083            RES2=TOT2(1,1)-REG2(1,1)                                       PSTT1240
 0084            MS2=DABS(RES2)/DFLOAT(IDF)                                     PSTT1250
          C                                                                     PSTT1260
          CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDPSTT1270
 0085            IF(IOUT.NE.-1) GO TO 60                                        PSTT1280
 0086            WRITE(IPRT,710) MSRES,MSREG,TVAL,FVAL,PROB                     PSTT1290
 0087       710  FORMAT(1H0,6HMSRES=,G15.8,3X,6HMSREG=,G15.8,3X,                PSTT1300
               A5HTVAL=,G15.8,3X,5HFVAL=,G15.8,3X,5HPROB=,G15.8,//)             PSTT1310
 0088            DO 40 J=1,25                                                   PSTT1320
 0089            WRITE(IPRT,720)(WT(I,J),I=1,10)                                PSTT1330
 0090      40    CONTINUE                                                       PSTT1340
 0091       720  FORMAT(1H0,2HWT,3X,10(G12.5))                                  PSTT1350
 0092            DO 50 I=1,10                                                   PSTT1360
 0093            WRITE(IPRT,730)(R2(I,J),J=1,10)                                PSTT1370
 0094      50    CONTINUE                                                       PSTT1380
 0095       730  FORMAT(1H ,2HR2,3X,10(G12.5),/)                                PSTT1390
 0096            WRITE(IPRT,740) REG2(1,1),RES2,TOT2(1,1),MS2                   PSTT1400
 0097       740  FORMAT(1H0,5HREG2=,G15.8,5X,5HRES2=,G15.8,5X,                  PSTT1410
               A5HTOT2=,G15.8,5X,4HMS2=,G15.8,/)                                PSTT1420
 0098      60    CONTINUE                                                       PSTT1430
          CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDPSTT1440
          C                                                                     PSTT1450
 0099                CALL LINV1F(R2,NCW,10,R2I,10,WORK,IER)                     PSTT1460
 0100                CALL DSMPY(R2I,MS2,VARB,10,10,0,100)                       PSTT1470
          C                                                                     PSTT1480
          CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDPSTT1490
 0101            IF(IOUT.NE.-1) GO TO 90                                        PSTT1500
 0102            DO 70 I=1,10                                                   PSTT1510
 0103            WRITE(IPRT,750)(R2I(I,J),J=1,10)                               PSTT1520
 0104      70    CONTINUE                                                       PSTT1530
 0105       750  FORMAT(1H ,3HR2I,2X,10(G12.5),/)                               PSTT1540
 0106            DO 80 I=1,10                                                   PSTT1550
 0107            WRITE(IPRT,760)(VARB(I,J),J=1,10)                              PSTT1560
 0108      80    CONTINUE                                                       PSTT1570
 0109       760  FORMAT(1H ,4HVA-B,1X,10(G12.5),/)                              PSTT1580
 0110      90    CONTINUE                                                       PSTT1590
          CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDPSTT1600
          C                                                                     PSTT1610
          C    LOOP 100 DETERMINES THE VARIANCE AND CONFIDENCE INTERVAL         PSTT1620
          C     FOR THE YI SOLUTIONS FOR A GIVEN RISK OF ALPHA                  PSTT1630
 0111                  DO 100 I=1,NCW                                           PSTT1640
 0112                  VARYI(I)=VARB(I,I)*(PB(I)**2)                            PSTT1650
 0113                  LIMYI(I)=TVAL*DSQRT(VARYI(I))                            PSTT1660
 0114       100        CONTINUE                                                 PSTT1670
          C                                                                     PSTT1680
          C                                                                     PSTT1690
          C    LOOP 108 DETERMINES THE ESTIMATED STANDARD ERROR                 PSTT1700
          C      AND THE CONFIDENCE INTERVAL (FOR GIVEN ALPHA)                  PSTT1710
          C     FOR THE ESTIMATED BOUNDARY CONDITIONS                           PSTT1720
 0115                  DO 108 I=1,NRW                                           PSTT1730
```

```
                  DO 105 J=1,NCW
                  WK(1,J)=W(I,J)
                  WKT(J,1)=W(I,J)
C
CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
      IF(IOUT.NE.-1) GO TO 105
      WRITE(IPRT,762)WK(1,J),WKT(J,1)
  762 FORMAT(1H ,2HWK,3X,G15.8,10X,3HWKT,2X,G15.8)
CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
C
  105             CONTINUE
            CALL DMPRD(WK,VARB,R3,1,10,0,0,10)
            CALL DMPRD(R3,WKT,VARD,1,10,0,0,1)
            SEBC(I)=DSQRT(VARD(1,1))
            TERM=TVAL*SEBC(I)
            BCLL(I)=ZHAT(I,1)-TERM
            BCUL(I)=ZHAT(I,1)+TERM
            RSDUL(I)=ZEE(I,1)-ZHAT(I,1)
            SMRES=SMRES+RSDUL(I)
C
CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
      IF(IOUT.NE.-1) GO TO 108
      WRITE(IPRT,764)(R3(1,K),K=1,10)
  764 FORMAT(1H0,2HR3,3X,10G12.5)
      WRITE(IPRT,766)VARD(1,1),TERM
  766 FORMAT(1H0,4HVARD,1X,G15.8,10X,4HTERM,1X,G15.8)
CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
C
  108       CONTINUE
C-------------------------------------------------------------------------
C-------------------------------------------------------------------------
C   OUTPUT OF SUBROUTINE PSTAT
C
      WRITE(IPRT,770)
  770 FORMAT(1H1,2(125(1H*)/),5X,53HFOLLOWING IS THE OUTPUT OF THE POST
     1STATISTICAL STUDY)
      WRITE(IPRT,780)
  780 FORMAT(1H0,10X,6HSOURCE,10X,14HSUM OF SQUARES,10X,
     118HDEGREES OF FREEDOM,10X,11HMEAN SQUARE,/,11X,6(1H-),10X,
     214(1H-),10X,13(1H-),10X,11(1H-))
      WRITE(IPRT,790)REG(1,1),NCW,MSREG,RES,IDF,MSRES,TOT(1,1),NRZ
  790 FORMAT(1H ,10X,10HREGRESSION,6X,G15.8,9X,I5,23X,G15.8,/,11X,
     A8HRESIDUAL,8X,G15.8,9X,I5,23X,G15.8,7H = S**2,//,
     B11X,10HTOTAL(UNC),6X,G15.8,9X,I5)
      WRITE(IPRT,800)RSQD,CV
  800 FORMAT(1H0,/,5X,18HPER-CENT VARIATION,3X,5HR**2=,G15.8,1H ,
     A5X,24HCOEFFICIENT OF VARIATION,3X,3HCV=,G15.8)
      WRITE(IPRT,810)FCAL,FVAL,PROB,ALPHA
  810 FORMAT(1H0,4X,26HTEST OF OVERALL REGRESSION,2X,5HFCAL=,
     AG15.8,2X,9HF(ALPHA)=,G15.8,2X,20HP(F(ALPHA).GT.FCAL)=,
     BG15.8,/,10X,18HFOR RISK OF ALPHA=,G15.8)
      IF(FCAL-FVAL)110,110,120
  110 WRITE(IPRT,820)
  820 FORMAT(1H ,09X,27H*****REJECT REGRESSION*****)
      GO TO 130
  120 WRITE(IPRT,830)
  830 FORMAT(1H ,09X,27H*****ACCEPT REGRESSION*****)
  130 CONTINUE
```

```
0154          WRITE(IPRT,840) TVAL,ALPHA                                              PSTT2320
0155     840  FORMAT(1H0,4X,36HSPECIFICS OF THE BOUNDARY CONDITIONS,                  PSTT2330
             A4X,10HT(ALPHA)= ,G15.8,4X,7HALPHA= ,G15.8,/,10X,                        PSTT2340
             B15HOBSERVED VALUES,3X,15HESTIMATED VALUE,5X,8HRESIDUAL,7X,              PSTT2350
             C14HESTIMATED S.E.,6X,11HLOWER LIMIT,07X,11HUPPER LIMIT)                 PSTT2360
0156          WRITE(IPRT,850)(ZEE(I,1),ZHAT(I,1),RSDUL(I),SEBC(I),                    PSTT2370
             1BCLL(I),BCUL(I),I=1,NRZ)                                                PSTT2380
0157     850  FORMAT(1H ,9X,G15.8,3X,G15.8,3X,G15.8,3X,G15.8,3X,G15.8,3X,G15.8) PSTT2390
0158          WRITE(IPRT,860) ZBAR,SMRES                                              PSTT2400
0159     860  FORMAT(1H ,9X,21HMEAN OF OBSERVATIONS=,G15.8,/,10X,                     PSTT2410
             A21HSUM OF THE RESIDUALS=,G15.8)                                         PSTT2420
0160          WRITE(IPRT,870)TVAL,ALPHA                                               PSTT2430
0161     870  FORMAT(1H0,5X,29HSPECIFICS OF THE YI SOLUTIONS,                         PSTT2440
             A4X,10HT(ALPHA)= ,G15.8,4X,7HALPHA= ,G15.8,/,10X,                        PSTT2450
             B15HESTIMATED VALUE,7X,12HEST VARIANCE,13X,19HCONFIDENCE INTERVAL) PSTT2460
0162          WRITE(IPRT,880)(YIR(I),VARYI(I),YIR(I),LIMYI(I),I=1,NCW)                PSTT2470
0163     880  FORMAT(1H ,10X,G15.8,5X,G15.8,5X,G15.8,6H (+-) ,G15.8)                  PSTT2480
0164          WRITE(IPRT,890)                                                         PSTT2490
0165     890  FORMAT(1H0,5X,48HCOVARIANCE MATRIX OF THE SUPERPOSITION CONSTANTS)PSTT2500
0166          DO 140 I=1,NCW                                                          PSTT2510
0167          WRITE(IPRT,900)(VARB(I,J),J=1,NCW)                                      PSTT2520
0168     140  CONTINUE                                                                PSTT2530
0169     900  FORMAT(1H ,2X,10(G12.5,1X))                                             PSTT2540
0170          WRITE(IPRT,910)                                                         PSTT2550
0171     910  FORMAT(1H0,4X,22HCONCLUDES STAT PACKAGE,/,2(125(1H*)/))                 PSTT2560
        C--------------------------------------------------------------------------PSTT2570
0172          RETURN                                                                  PSTT2580
0173          END                                                                     PSTT2590
```

```
FORTRAN IV G LEVEL  21                PSTAT             DATE = 76097        15/49/10            PAGE 0006

  *OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
  *OPTIONS IN EFFECT*  NAME = PSTAT   , LINECNT =       60
  *STATISTICS*    SOURCE STATEMENTS =       173,PROGRAM SIZE =    12044
  *STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
0001          SUBROUTINE DMATA(A,R,N,M,MS)                                DMATA000
0002          IMPLICIT REAL*8(A-H,O-Z)                                    DMATA010
0003          REAL*8     A(1),R(1)                                        DMATA020
        C-----------------------------------------------------------------DMATA030
        C    DMATA PREMULTIPLIES A MATRIX BY ITS TRANSPOSE                DMATA040
        C     TO FORM A RESULTANT MATRIX                                  DMATA050
        C                                                                 DMATA060
        C    A IS THE INPUT MATRIX                                        DMATA070
        C    R IS THE OUTPUT MATRIX                                       DMATA080
        C    N IS THE NUMBER OF ROWS IN A                                 DMATA090
        C    M IS THE NUMBER OF COLUMNS IN A.  ALSO THE NUMBER            DMATA100
        C     OF ROWS AND COLUMNS OF R                                    DMATA110
        C    MS IS A ONE DIGIT NUMBER FOR THE STORAGE MODE OF MATRIX A    DMATA120
        C     0-GENERAL  1-SYMETRICAL  2-DIAGONAL                         DMATA130
        C-----------------------------------------------------------------DMATA140
0004          DO 60 K=1,M                                                 DMATA150
0005          KX=(K*K-K)/2                                                DMATA160
0006          DO 60 J=1,M                                                 DMATA170
0007          IF(J-K) 10,10,60                                            DMATA180
0008       10 IR=J+KX                                                     DMATA190
0009          R(IR)=0.0D0                                                 DMATA200
0010          DO 60 I=1,N                                                 DMATA210
0011          IF(MS) 20,40,20                                             DMATA220
0012       20 CALL LOC(I,J,IA,N,M,MS)                                     DMATA230
0013          CALL LOC(I,K,IB,N,M,MS)                                     DMATA240
0014          IF(IA) 30,60,30                                             DMATA250
0015       30 IF(IB) 50,60,50                                             DMATA260
0016       40 IA=N*(J-1)+I                                                DMATA270
0017          IB=N*(K-1)+I                                                DMATA280
0018       50 R(IR)=R(IR)+A(IA)*A(IB)                                     DMATA290
0019       60 CONTINUE                                                    DMATA300
0020          RETURN                                                      DMATA310
0021          END                                                         DMATA320
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = DMATA    , LINECNT =       60
*STATISTICS*    SOURCE STATEMENTS =       21,PROGRAM SIZE =       840
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
FORTRAN IV G LEVEL  21                DMCPY             DATE = 76097          15/49/10              PAGE 0001

0001              SUBROUTINE DMCPY(A,R,N,M,MS)                                        DMCPY000
0002              IMPLICIT REAL*8(A-H,O-Z)                                            DMCPY010
0003              REAL*8 A(1),R(1)                                                    DMCPY020
            C-------------------------------------------------------------------------DMCPY030
            C    DMCPY COPIES AN ENTIRE MATRIX (USED BY DMTRA)                        DMCPY040
            C                                                                         DMCPY050
            C    A IS THE INPUT MATRIX                                                DMCPY060
            C    R IS THE OUTPUT MATRIX                                               DMCPY070
            C    N IS THE NUMBER OF ROWS IN A AND R                                   DMCPY080
            C    M IS THE NUMBER OF COLUMNS IN A AND R                                DMCPY090
            C    MS IS A ONE DIGIT NUMBER FOR THE STORAGE MODE OF A(AND R)            DMCPY100
            C     0-GENERAL  1-SYMETRICAL  2-DIAGONAL                                 DMCPY110
            C-------------------------------------------------------------------------DMCPY120
            C    COMPUTE VECTOR LENGTH,IT                                             DMCPY130
0004              CALL LOC(N,M,IT,N,M,MS)                                             DMCPY140
            C    COPY MATRIX                                                          DMCPY150
0005              DO1 I=1,IT                                                          DMCPY160
0006          1   R(I)=A(I)                                                           DMCPY170
0007              RETURN                                                              DMCPY180
0008              END                                                                 DMCPY190
```

```
FORTRAN IV G LEVEL  21                DMCPY              DATE =   097          15/49/10              PAGE 0002

 *OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*  NAME = DMCPY    , LINECNT =       60
 *STATISTICS*     SOURCE STATEMENTS =        8,PROGRAM SIZE =      478
 *STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
ORTRAN IV G LEVEL  21              DMPRD              DATE = 76097          15/49/10              PAGE 0001

0001          SUBROUTINE DMPRD(A,B,R,N,M,MSA,MSB,L)                                   DMPRD000
0002          IMPLICIT REAL*8(A-H,O-Z)                                                DMPRD010
0003          REAL*8     A(1),B(1),R(1)                                               DMPRD020
        C-----------------------------------------------------------------------------DMPRD030
        C   DMPRD MULTIPLIES TO MATRICES TO FORM A RESULTANT MATRIX                   DMPRD040
        C                                                                             DMPRD050
        C   A IS THE FIRST INPUT MATRIX                                               DMPRD060
        C   B IS THE SECOND INPUT MATRIX                                              DMPRD070
        C   R IS THE OUTPUT MATRIX                                                    DMPRD080
        C   N IS THE NUMBER OF ROWS IN A AND R                                        DMPRD090
        C   M IS THE NUMBER OF COLUMNS IN A AND ROWS IN B                             DMPRD100
        C   MSA IS A ONE DIGIT NUMBER FOR THE STORAGE MODE OF A                       DMPRD110
        C   MSB IS A ONE DIGIT NUMBER FOR THE STORAGE MODE OF B                       DMPRD120
        C    0-GENERAL  1-SYMETRICAL  2-DIAGONAL                                      DMPRD130
        C   L IS THE NUMBER OF COLUMNS IN B AND R                                     DMPRD140
        C-----------------------------------------------------------------------------DMPRD150
        C        SPECIAL CASE FOR DIAGONAL BY DIAGONAL                                DMPRD160
0004          MS=MSA*10+MSB                                                           DMPRD170
0005          IF(MS-22) 30,10,30                                                      DMPRD180
0006       10 DO 20 I=1,N                                                             DMPRD190
0007       20 R(I)=A(I)*B(I)                                                          DMPRD200
0008          RETURN                                                                  DMPRD210
        C        ALL OTHER CASES                                                      DMPRD220
0009       30 IR=1                                                                    DMPRD230
0010          DO 90 K=1,L                                                             DMPRD240
0011          DO 90 J=1,N                                                             DMPRD250
0012          R(IR)=0.0D0                                                             DMPRD260
0013          DO 80 I=1,M                                                             DMPRD270
0014          IF(MS) 40,60,40                                                         DMPRD280
0015       40 CALL LOC(J,I,IA,N,M,MSA)                                                DMPRD290
0016          CALL LOC(I,K,IB,M,L,MSB)                                                DMPRD300
0017          IF(IA) 50,80,50                                                         DMPRD310
0018       50 IF(IB) 70,80,70                                                         DMPRD320
0019       60 IA=N*(I-1)+J                                                            DMPRD330
0020          IB=M*(K-1)+I                                                            DMPRD340
0021       70 R(IR)=R(IR)+A(IA)*B(IB)                                                 DMPRD350
0022       80 CONTINUE                                                                DMPRD360
0023       90 IR=IR+1                                                                 DMPRD370
0024          RETURN                                                                  DMPRD380
0025          END                                                                     DMPRD390
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = DMPRD    , LINECNT =       60
*STATISTICS*     SOURCE STATEMENTS =        25,PROGRAM SIZE =      1010
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
      SUBROUTINE DMTRA(A,R,N,M,MS)                                      DMTRA000
      IMPLICIT REAL*8(A-H,O-Z)                                          DMTRA010
      REAL*8    A(1),R(1)                                               DMTRA020
C-----------------------------------------------------------------------DMTRA030
C   DMTRA DETERMINES THE TRANSPOSE OF A MATRIX                          DMTRA040
C                                                                       DMTRA050
C   A IS THE MATRIX TO BE TRANSPOSED                                    DMTRA060
C   R IS THE TRANSPOSED MATRIX                                          DMTRA070
C   N IS THE NUMBER OF ROWS OF A AND COLUMNS OF R                       DMTRA080
C   M IS THE NUMBER OF COLUMNS OF A AND ROWS OF R                       DMTRA090
C   MS IS ONE DIGIT NUMBER FOR STORAGE MODE OF MATRIX A(AND R)          DMTRA100
C    0-GENERAL  1-SYMETRICAL  2-DIAGONAL                                DMTRA110
C-----------------------------------------------------------------------DMTRA120
C          IF MS IS 1 OR 2, COPY A                                      DMTRA130
      IF(MS) 10,20,10                                                   DMTRA140
   10 CALL DMCPY(A,R,N,N,MS)                                            DMTRA150
      RETURN                                                            DMTRA160
C          TRANSPOSE GENERAL MATRIX                                     DMTRA170
   20 IR=0                                                              DMTRA180
      DO 30 I=1,N                                                       DMTRA190
      IJ=I-N                                                            DMTRA200
      DO 30 J=1,M                                                       DMTRA210
      IJ=IJ+N                                                           DMTRA220
      IR=IR+1                                                           DMTRA230
   30 R(IR)=A(IJ)                                                       DMTRA240
      RETURN                                                            DMTRA250
      END                                                               DMTRA260
```

```
... ... .. U LEVEL  21              DMTRA             DATE = 76097        15/49/10          PAGE 0002

*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = DMTRA    , LINECNT =       60
*STATISTICS*    SOURCE STATEMENTS =       15,PROGRAM SIZE =      618
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
                              DSMPY            DATE = 76097        15/49/10            PAGE 0001
      SUBROUTINE DSMPY(A,C,R,N,M,MS,IT)                                DSMPY000
      IMPLICIT REAL*8(A-H,O-Z)                                         DSMPY010
      REAL*8     A(1),R(1)                                             DSMPY020
C----------------------------------------------------------------------DSMPY030
C    DSMPY MULTIPLIES EACH ELEMENT OF A MATRIX BY A SCALER TO          DSMPY040
C     FORM A RESULTANT MATRIX                                          DSMPY050
C                                                                      DSMPY060
C    A IS THE INPUT MATRIX                                             DSMPY070
C    C IS THE SCALER                                                   DSMPY080
C    R IS THE OUTPUT MATRIX                                            DSMPY090
C    N IS THE NUMBER OF ROWS IN MATRIX A AND R                         DSMPY100
C    M IS THE NUMBER OF COLUMNS IN A AND R                             DSMPY110
C    MS IS A ONE DIGIT NUMBER FOR STORAGE MODE OF A(AND R)             DSMPY120
C     0-GENERAL  1-SYMETRICAL  2-DIAGONAL                              DSMPY130
C    IT IS THE TOTAL LENGTH OF VECTOR A AND R                          DSMPY140
C----------------------------------------------------------------------DSMPY150
C         MULTIPLY BY SCALAR                                           DSMPY160
      DO 1 I=1,IT                                                      DSMPY170
    1 R(I)=A(I)*C                                                      DSMPY180
      RETURN                                                           DSMPY190
      END                                                              DSMPY200
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = DSMPY    , LINECNT =       60
*STATISTICS*    SOURCE STATEMENTS =        7,PROGRAM SIZE =       492
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
0001        SUBROUTINE DTPRD(A,B,R,N,M,MSA,MSB,L)                              DTPRD000
0002        IMPLICIT REAL*8(A-H,O-Z)                                           DTPRD010
0003        REAL*8 A(1),B(1),R(1)                                              DTPRD020
      C-----------------------------------------------------------------------DTPRD030
      C    DTPRD TRANSPOSES A MATRIX AND POSTMULTIPLIES BY                    DTPRD040
      C     ANOTHER TO FORM A RESULTANT MATRIX                                DTPRD050
      C                                                                       DTPRD060
      C    A IS THE FIRST INPUT MATRIX(TO BE TRANSPOSED)                      DTPRD070
      C    B IS THE SECOND INPUT MATRIX                                       DTPRD080
      C    R IS THE OUTPUT MATRIX                                             DTPRD090
      C    N IS THE NUMBER OF ROWS IN A AND B                                 DTPRD100
      C    M IS THE NUMBER OF COLUMNS IN A AND ROWS IN R                      DTPRD110
      C    MSA IS A ONE DIGIT NUMBER FOR THE STORAGE MODE OF MATRIX A         DTPRD120
      C    MSB IS A ONE DIGIT NUMBER FOR THE STORAGE MODE OF MATRIX B         DTPRD130
      C     0-GENERAL  1-SYMETRICAL  2-DIAGONAL                               DTPRD140
      C    L IS THE NUMBER OF COLUMNS IN B AND R                              DTPRD150
      C-----------------------------------------------------------------------DTPRD160
      C    SPECIAL CASE FOR DIAGONAL BY DIAGOMAL                              DTPRD170
0004        MS=MSA*10+MSB                                                      DTPRD180
0005        IF(MS-22) 30,10,30                                                 DTPRD190
0006   10   DO 20 I=1,N                                                        DTPRD200
0007   20   R(I)=A(I)*B(I)                                                     DTPRD210
0008        RETURN                                                             DTPRD220
      C    MULTIPLY TRANSPOSE OF A BY B                                       DTPRD230
0009   30   IR=1                                                               DTPRD240
0010        DO 90 K=1,L                                                        DTPRD250
0011        DO 90 J=1,M                                                        DTPRD260
0012        R(IR)=0.0D0                                                        DTPRD270
0013        DO 80 I=1,N                                                        DTPRD280
0014        IF(MS) 40,60,40                                                    DTPRD290
0015   40     CALL LOC(I,J,IA,N,M,MSA)                                         DTPRD300
0016        CALL LOC(I,K,IB,N,L,MSB)                                           DTPRD310
0017        IF(IA) 50,80,50                                                    DTPRD320
0018   50   IF(IB) 70,80,70                                                    DTPRD330
0019   60   IA=N*(J-1)+I                                                       DTPRD340
0020        IB=N*(K-1)+I                                                       DTPRD350
0021   70   R(IR)=R(IR)+A(IA)*B(IB)                                            DTPRD360
0022   80   CONTINUE                                                           DTPRD370
0023   90   IR=IR+1                                                            DTPRD380
0024        RETURN                                                             DTPRD390
0025        END                                                                DTPRD400
```

```
*OPTIONS IN EFFECT*  ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*  NAME = DTPRD    , LINECNT =       60
*STATISTICS*    SOURCE STATEMENTS =        25,PROGRAM SIZE =      1010
*STATISTICS*  NO DIAGNOSTICS GENERATED

*STATISTICS*  NO DIAGNOSTICS THIS STEP
```

VS LOADER

TIONS USED - PRINT,MAP,NOLET,CALL,RES,NOTERM,SIZE=159744,NAME=**GO

NAME	TYPE	ADDR	NAME	TYPE	ADDR	NAME	TYPE	ADDR	NAME	TYPE	ADDR	NAME	TYPE	ADDR
MAIN	SD	4E1010	PYPS	SD	4E4F00	REG	SD	4E9848	DTFIND	SD	4EA368	DEVAL	SD	4EA6E8
EVAL	SD	4EAA48	GJRWLS	SD	4EAD70	LSTSQS	SD	4EBB80	CALCSR	SD	4EC168	RECURL	SD	4EC408
TRIGF	SD	4ECA28	PSTAT	SD	4ECF30	DMATA	SD	4EFE40	DMCPY	SD	4F0188	DMPRD	SD	4F0368
DMTRA	SD	4F0760	DSMPY	SD	4F09D0	DTPRD	SD	4F0BC0	LOC **	SD	EC5934	IHCECOMH*	SD	4F0F88
IBCOM *	LR	4F0F88	FDIOCS *	LR	4F1074	INTSWTCH*	LR	4F1EFE	IHCCOMH2*	SD	4F1F20	SEQDASD *	LR	4F2298
IHCFEXIT*	SD	4F2580	EXIT *	LR	4F2580	IHCLEXP *	SD	4F25A0	DEXP *	LR	4F25A0	IHCLLOG *	SD	4F2828
DLOG10 *	LR	4F2828	DLOG *	LR	4F2840	IHCLSCN *	SD	4F2A28	DCOS *	LR	4F2A28	DSIN *	LR	4F2A42
MOFI *	SD	4F2C98	MDSTI *	SD	4F2E80	MDFD *	SD	4F33F8	LINV1F *	SD	4F3D58	IHCLSQRT*	SD	4F3FF0
DSQRT *	LR	4F3FF0	IHCFCVTH*	SD	4F4150	ADCON *	LR	4F4150	FCVAOUTP*	LR	4F41FA	FCVLOUTP*	LR	4F428A
FCVZOUTP*	LR	4F43E2	FCVIOUTP*	LR	4F4796	FCVEOUTP*	LR	4F4C98	FCVCOUTP*	LR	4F4EB2	INT6SWCH*	LR	4F5198
IHCEFIOS*	SD	4F5308	FIOCS *	LR	4F5308	FIOCSBEP*	LR	4F530E	IHCFIOS2*	SD	4F6230	IHCEFNTH*	SD	4F6760
ARITH *	LR	4F6760	ADJSWTCH*	LR	4F6AFC	IHCUOPT *	SD	4F6CA8	IHCERRM *	SD	4F6FA8	ERRMON *	LR	4F6FA8
IHCERRE *	LR	4F6FC0	MDBETI *	SD	4F7588	UERTST *	SD	4F7818	IHCSEXP *	SD	4F7A90	EXP *	LR	4F7A90
IHCFRXPR*	SD	4F7C28	FRXPR *	LR	4F7C28	IHCSTNCT*	SD	4F7D30	COTAN *	LR	4F7DB0	TAN *	LR	4F7DC6
CTAN *	LR	4F7F38	IHCSSQRT*	SD	4F8018	SQRT *	LR	4F8018	MERFI *	SD	4F8160	MDNRIS *	LR	4F8174
MERFCI *	LR	4F818A	IHCSATN2*	SD	4F8470	ATAN2 *	LR	4F8470	ATAN *	LR	4F8484	IHCSERF *	SD	4F8640
ERFC *	LR	4F8640	ERF *	LR	4F8656	IHCSLOG *	SD	4F8830	ALOG10 *	LR	4F8830	ALOG *	LR	4F8848
LEQT1F *	SD	4F89E8	IHCUATBL*	SD	4F8C90	IHCETRCH*	SD	4F92C8	IHCTRCH *	LR	4F92C8	ERRTRA *	LR	4F92D0
MDBETA *	SD	4F9558	LUDATF *	SD	4F9CB0	LUELMF *	SD	4FA6A0	IHCLGAMA*	SD	4FAA58	DLGAMA *	LR	4FAA58
DGAMMA *	LR	4FAA74	IHCFDXPI*	SD	4FAE48	FDXPI *	LR	4FAE48	LOG	CM	4FAF98	PARM	CM	4FAFA0

TOTAL LENGTH 19FB4
ENTRY ADDRESS 4E1010

```
****************************************************************************************************
****************************************************************************************************
FOLLOWING IS THE OUTPUT OF INPUT

LINEAR=     0                                   STAT=    0           ALPHA= 0.0
    LINEAR= ZERO IMLIES NON-LINEAR                  STAT= ONE IMPLIES POST STATISTICAL STUDY TO BE EXECUTED
    LINEAR= NON-ZERO IMPLIES LINEAR                 STAT= ZERO IMPLIES NO POST SATISTICAL STUDY

ACCURACY                           ACC= 0.10000D-05
FOR GJWRIS                         CHECK= 0.10000D-19     DET=  1.0000
FUDGE FACTOR FOR DTFIND            RF= 0.50000

NUMBER OF DIFFERENTIAL EQ        NDE=        4
NUMBER OF BOUNDARY CONDITIONS    NBV=       15
NUMBER OF TERMS IN PWR SERIES    NTERMS=    20
NO. EXACT BOUNDARY CONDITIONS    NEMAX=      1
NO. CF NCN-TRIVIAL DIFF EQ       NEQ=        2

BOUNDARY VALUES     APPLIED ON(IBV)       IQBV       TIME OF BV
-0.22000000                 -2             1          1.0000000
 0.35000000D-01             -2             1          2.0000000
-0.47400000                 -2             1          3.0000000
-0.58900000                 -2             1          4.0000000
 0.39300000                 -2             1          5.0000000
  1.5970000                 -2             1          6.0000000
  1.4520000                 -2             1          7.0000000
-0.38300000                 -2             1          8.0000000
 -2.3240000                 -2             1          9.0000000
 -2.2740000                 -2             1          10.000000
 0.88000000D-01             -2             1          11.000000
  2.7110000                 -2             1          12.000000
  2.9970000                 -2             1          13.000000
 0.40100000                 -2             1          14.000000
 -2.8160000                 -2             1          15.000000

INITIAL TIME OF INTEGRATION      TO= 0.0

INITIAL GUESSES OF Y                        ICEX        LOWER LIMIT         UPPER LIMIT
     Y 1  YI( 1)= 0.50000000                 1            0.0                 0.0
     Y 2  YI( 2)=  1.0000000                 1            0.0                 0.0
     Y 3  YI( 3)= 0.10000000D 00             1            0.0                 0.0
     Y 4  YI( 4)=  1.3000000                 1            0.0                 0.0

PERTURBATION SCALER             PTBS= 0.10000000D 00
    ABSOLUTE VALUE LIMITS       PTMIN= 0.0
                                PTMAX=  2.0000

IOUT=    0
QUIT=   10

CONCLUDES FORMAL OUTPUT OF THE INPUT
****************************************************************************************************
****************************************************************************************************
```

```
ITERATION    1

S MATRIX(CONTAINS UNPERTERBED AND PERTURBED SOL)
      1.0000          1.0000          1.0000          1.0000          1.0000          1.0000
     -0.64228        -0.66568        -0.74448        -0.64035        -0.72161        -0.22000
     -0.49241        -0.45223        -0.56472        -0.48344        -0.42572         0.35000D-01
     -0.51667        -0.46350        -0.48175        -0.51154        -0.33497        -0.47400
     -0.68726D-01    -0.62761D-01     0.24495D-01    -0.74149D-01     0.67776D-01    -0.58900
      1.0073          0.96399         1.0500          0.99151         0.95145         0.39300
      1.6095          1.5696          1.5591          1.5937          1.3335          1.5970
      0.58079         0.58825         0.50209         0.58413         0.25038         1.4520
     -1.5115         -1.4695         -1.5285         -1.4801         -1.5760         -0.38800
     -2.4989         -2.4722         -2.4413         -2.4623         -2.0932         -2.3240
     -0.93638        -0.95315        -0.87507        -0.93819        -0.30496        -2.2740
      1.9457          1.9082          1.9424          1.8904          2.1686          0.88000D-01
      3.1756          3.1609          3.1175          3.1139          2.5807          2.7110
      1.1797          1.2019          1.1364          1.1841          0.18644         2.9970
     -2.2683         -2.2373         -2.2501         -2.1834         -2.6331          0.40100
     -3.6462         -3.6417         -3.5926         -3.5601         -2.8052         -2.8160

SUPERPOSITION CONSTANTS
     A( 1) =  -23.496468
     A( 2) =    4.5826473
     A( 3) =   -3.2130453
     A( 4) =   24.602487
     A( 5) =   -1.4756211

SOLUTION FOR STATE VARIABLES
     TRUE IC OF Y( 1) IS  0.72913236
     TRUE IC OF Y( 2) IS  0.67869547
     TRUE IC OF Y( 3) IS  0.34602487
     TRUE IC OF Y( 4) IS   1.1081693
```

```
ITERATION    2

S MATRIX(CONTAINS UNPERTERBED AND PERTURBED SOL)
      1.0000        1.0000        1.0000        1.0000        1.0000        1.0000
    -0.28400      -0.30875      -0.34368      -0.27680      -0.32611      -0.22000
    -0.10934     -0.73301D-01   -0.14369      -0.10041     -0.60240D-01   0.35000D-01
    -0.52390      -0.47562      -0.51101      -0.51597      -0.43741      -0.47400
    -0.44776      -0.43203      -0.41245      -0.44100      -0.36722      -0.58900
     0.53576       0.51504       0.55679       0.52034       0.57264       0.39300
      1.4436        1.4148        1.4366        1.3975        1.3700        1.5970
      1.0295        1.0196        1.0087       0.99392       0.82706        1.4520
    -0.62154      -0.60964      -0.63440      -0.58872      -0.81733      -0.38800
     -1.9529       -1.9357       -1.9491       -1.8563       -1.9219       -2.3240
     -1.4983       -1.4921       -1.4860       -1.4271       -1.1795       -2.2740
     0.50384       0.49703       0.51169       0.45960       0.86618      0.88000D-01
      2.1947        2.1844        2.1927        2.0530        2.2326        2.7110
      1.8772        1.8732        1.8699        1.7689        1.4797        2.9970
    -0.26560      -0.26170      -0.27038      -0.22084      -0.77395       0.40100
     -2.2557       -2.2496       -2.2547       -2.0810       -2.3812       -2.8160

SUPERPOSITION CONSTANTS
     A( 1) =    3.8263156
     A( 2) =    5.2942813
     A( 3) =   -2.1706495
     A( 4) =   -4.3846907
     A( 5) =   -1.5652568

SOLUTION FOR STATE VARIABLES
     TRUE IC OF Y( 1) IS    1.1151556
     TRUE IC OF Y( 2) IS   0.53137447
     TRUE IC OF Y( 3) IS   0.19430367
     TRUE IC OF Y( 4) IS   0.93471231
```

ITERATION 3

S MATRIX(CONTAINS UNPERTERBED AND PERTURBED SOL)

1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
-0.26299	-0.30925	-0.30601	-0.25780	-0.31348	-0.22000
0.15821D-01	0.53629D-01	-0.20329D-01	0.19082D-01	0.73683D-01	0.35000D-01
-0.46096	-0.38362	-0.46306	-0.45911	-0.35464	-0.47400
-0.58687	-0.53773	-0.55929	-0.58120	-0.48996	-0.58900
0.36125	0.34858	0.39161	0.35862	0.41187	0.39300
1.5921	1.5385	1.6009	1.5649	1.5326	1.5970
1.5573	1.5121	1.5414	1.5201	1.3369	1.4520
-0.20990	-0.21269	-0.23361	-0.21022	-0.49802	-0.38800
-2.2593	-2.2249	-2.2708	-2.1949	-2.3554	-2.3240
-2.4772	-2.4392	-2.4696	-2.3899	-2.1736	-2.2740
-0.28135	-0.27024	-0.26402	-0.26147	0.28847	0.88000D-01
2.5169	2.4971	2.5286	2.4212	2.8723	2.7110
3.2457	3.2161	3.2437	3.0989	2.9600	2.9970
0.96133	0.94683	0.94951	0.90198	0.14009	0.40100
-2.4385	-2.4292	-2.4491	-2.3286	-3.1189	-2.8160

SUPERPOSITION CONSTANTS
A(1) = 1.3517611
A(2) = -1.1611008
A(3) = -0.48508048
A(4) = 0.63423510
A(5) = 0.66018502

SOLUTION FOR STATE VARIABLES
TRUE IC OF Y(1) IS 0.98567475
TRUE IC OF Y(2) IS 0.50559853
TRUE IC OF Y(3) IS 0.20662709
TRUE IC OF Y(4) IS 0.99642062

```
ITERATION    4

S MATRIX(CONTAINS UNPERTERBED AND PERTURBED SOL)
      1.0000        1.0000        1.0000        1.0000        1.0000        1.0000
    -0.21557      -0.25624      -0.25822      -0.21059      -0.26090      -0.22000
  0.23854D-01   0.63661D-01  -0.96796D-02   0.26537D-01   0.81132D-01   0.35000D-01
    -0.48682      -0.41452      -0.48517      -0.48462      -0.38917      -0.47400
    -0.59250      -0.55364      -0.56361      -0.58564      -0.50506      -0.58900
     0.39681       0.37631       0.42395       0.39377       0.44332       0.39300
      1.5982        1.5463        1.6014        1.5686        1.5363        1.5970
      1.4510        1.4166        1.4321        1.4140        1.2293        1.4520
    -0.38023      -0.37196      -0.40146      -0.37416      -0.65396      -0.38800
     -2.3051       -2.2690       -2.3107       -2.2337       -2.3590       -2.3240
     -2.2628       -2.2339       -2.2510       -2.1797       -1.9157       -2.2740
  0.70663D-01   0.69730D-01   0.86804D-01   0.71959D-01    0.62427      0.88000D-01
      2.6704        2.6460        2.6767        2.5579        2.9215        2.7110
      2.9700        2.9468        2.9631        2.8314        2.5612        2.9970
     0.42242       0.41932       0.41048       0.39875      -0.40697       0.40100
     -2.7546       -2.7387       -2.7608       -2.6115       -3.2646       -2.8160

SUPERPOSITION CONSTANTS
     A( 1) =    1.2542553
     A( 2) =   0.14870376
     A( 3) =  -0.11856874
     A( 4) =  -0.32043791
     A( 5) =   0.36047558D-01

SOLUTION FOR STATE VARIABLES
     TRUE IC OF Y( 1) IS    1.0003321
     TRUE IC OF Y( 2) IS   0.49960372
     TRUE IC OF Y( 3) IS   0.20000597
     TRUE IC OF Y( 4) IS    1.0000125
```

ITERATION 5

S MATRIX(CONTAINS UNPERTERBED AND PERTURBED SOL)

1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
-0.21984	-0.26149	-0.26211	-0.21482	-0.26524	-0.22000
0.35067D-01	0.75921D-01	0.15970D-02	0.37547D-01	0.94762D-01	0.35000D-01
-0.47327	-0.39891	-0.47164	-0.47161	-0.37377	-0.47400
-0.58918	-0.54936	-0.56018	-0.58280	-0.50300	-0.58900
0.39301	0.37137	0.42026	0.39032	0.43622	0.39300
1.5971	1.5432	1.6002	1.5689	1.5326	1.5970
1.4520	1.4166	1.4327	1.4163	1.2294	1.4520
-0.38817	-0.37892	-0.40968	-0.38227	-0.66096	-0.38800
-2.3238	-2.2857	-2.3293	-2.2543	-2.3740	-2.3240
-2.2739	-2.2440	-2.2617	-2.1931	-1.9200	-2.2740
0.68237D-01	0.86581D-01	0.10476	0.88882D-01	0.64776	0.88000D-01
2.7113	2.6851	2.7175	2.6004	2.9593	2.7110
2.9971	2.9727	2.9897	2.8612	2.5730	2.9970
0.40068	0.39803	0.38830	0.37894	-0.44600	0.40100
-2.8164	-2.7990	-2.8225	-2.6736	-3.3272	-2.8160

SUPERPOSITION CONSTANTS

A(1) = 0.99774769
A(2) = -0.27960235D-02
A(3) = 0.35900412D-02
A(4) = 0.13099471D-02
A(5) = -0.45165360D-03

SOLUTION FOR STATE VARIABLES

TRUE IC OF Y(1) IS 1.0000524
TRUE IC OF Y(2) IS 0.49978308
TRUE IC OF Y(3) IS 0.20004417
TRUE IC OF Y(4) IS 0.99996731

```
ITERATION    6

S MATRIX(CONTAINS UNPERTERBED AND PERTURBED SOL)
     1.0000        1.0000        1.0000        1.0000        1.0000        1.0000
    -0.21984      -0.26149      -0.26213      -0.21483      -0.26525      -0.22000
     0.34810D-01   0.75648D-01   0.13302D-02   0.37295D-01   0.94467D-01   0.35000D-01
    -0.47351      -0.39918      -0.47189      -0.47184      -0.37402      -0.47400
    -0.58922      -0.54940      -0.56021      -0.58283      -0.50299      -0.58900
     0.39314       0.37152       0.42040       0.39044       0.43641       0.39300
     1.5972        1.5433        1.6004        1.5690        1.5328        1.5970
     1.4521        1.4166        1.4328        1.4163        1.2294        1.4520
    -0.38813      -0.37890      -0.40965      -0.38224      -0.66097      -0.38800
    -2.3238       -2.2857       -2.3292       -2.2543       -2.3740       -2.3240
    -2.2740       -2.2440       -2.2617       -2.1931       -1.9201       -2.2740
     0.88050D-01   0.86402D-01   0.10458       0.88699D-01   0.64758       0.88000D-01
     2.7110        2.6849        2.7173        2.6001        2.9592        2.7110
     2.9971        2.9726        2.9897        2.8611        2.5731        2.9970
     0.40098       0.39833       0.38860       0.37923      -0.44563       0.40100
    -2.8159       -2.7985       -2.8221       -2.6732       -3.3269       -2.8160

SUPERPOSITION CONSTANTS
     A( 1) =  0.99999970
     A( 2) = -0.43169882D-06
     A( 3) =  0.72216856D-06
     A( 4) =  0.72061237D-08
     A( 5) = -0.16488120D-08

SOLUTION FOR STATE VARIABLES
     TRUE IC OF Y( 1) IS   1.0000524
     TRUE IC OF Y( 2) IS  0.49978311
     TRUE IC OF Y( 3) IS  0.20004417
     TRUE IC OF Y( 4) IS  0.99996731

ITERATION  6 CONVERGED TO THE ACCURACY SPECIFIED FOR THE SOLUTION
**************************************************************************************************
**************************************************************************************************
```

```
********************** I T E M I Z E D   J O B   C O S T **********************
*                                                                             *
* BOX       JOB NAME            COMMENTS                   DATE    JOB NUMBER *
* RRA       HUNTER        HUNTER99999999999999                       3388     *
*                                                                             *
* I T E M                 QUANTITY          COST        PREVIOUS              *
*                                                        BALANCE     105.30   *
* CPU                    16.62  SECONDS      5.98                             *
* CARD INPUT              1424  CARDS        0.00                             *
* PRINTED OUTPUT            31  PAGES        0.00                             *
*                                                                             *
*                                                                             *
*                                                                             *
*                                                                             *
*                                                                             *
*                                                                             *
*                                                                             *
*                                                                             *
*                                                                             *
*                                                                             *
*                                                                             *
*                                                  TOTAL CHARGES       5.98   *
*                                                                             *
*                                                CURRENT BALANCE      99.32   *
*                                                                             *
*******************************************************************************


 XXXXXXXXXX   XXXXXXXXXX    XXXXXXXXX                           XXXXXXXXX   XXXXXXXXX   XXXXXXXXX   XXXXXXXXX
 XXXXXXXXXXX  XXXXXXXXXXX  XXXXXXXXXXX                         XXXXXXXXXXX XXXXXXXXXXX XXXXXXXXXXX XXXXXXXXXXX
 XX       XX  XX       XX  XX       XX                         XX       XX XX       XX XX       XX XX       XX
 XX       XX  XX       XX  XX       XX                                  XX          XX XX       XX XX       XX
 XX       XX  XX       XX  XX       XX                                  XX          XX  XX     XX   XX     XX
 XXXXXXXXXXX  XXXXXXXXXXX  XXXXXXXXXXX                              XXXX        XXXX     XXXXXXX     XXXXXXX
 XXXXXXXXXX   XXXXXXXXXX   XXXXXXXXXXX                              XXXX        XXXX     XXXXXXX     XXXXXXX
 XX    XX     XX    XX     XX       XX                                  XX          XX  XX     XX   XX     XX
 XX     XX    XX     XX    XX       XX                                  XX          XX XX       XX XX       XX
 XX      XX   XX      XX   XX       XX                         XX       XX XX       XX XX       XX XX       XX
 XX       XX  XX       XX  XX       XX                         XXXXXXXXXXX XXXXXXXXXXX XXXXXXXXXXX XXXXXXXXXXX
 XX        XX XX        XX XX       XX                          XXXXXXXXX   XXXXXXXXX   XXXXXXXXX   XXXXXXXXX


JOB 3388   HUNTER    ENDED   ON RM2.PR1   4.00.52 PM   6 APR 76   BOX RRA   HUNTER        HASP-II VERSION 4.0

JOB 3388   HUNTER    ENDED   ON RM2.PR1   4.00.52 PM   6 APR 76   BOX RRA   HUNTER        HASP-II VERSION 4.0

JOB 3388   HUNTER    ENDED   ON RM2.PR1   4.00.52 PM   6 APR 76   BOX RRA   HUNTER        HASP-II VERSION 4.0

JOB 3388   HUNTER    ENDED   ON RM2.PR1   4.00.52 PM   6 APR 76   BOX RRA   HUNTER        HASP-II VERSION 4.0

JOB 3388   HUNTER    ENDED   ON RM2.PR1   4.00.52 PM   6 APR 76   BOX RRA   HUNTER        HASP-II VERSION 4.0

JOB 3388   HUNTER    ENDED   ON RM2.PR1   4.00.52 PM   6 APR 76   BOX RRA   HUNTER        HASP-II VERSION 4.0

JOB 3388   HUNTER    ENDED   ON RM2.PR1   4.00.52 PM   6 APR 76   BOX RRA   HUNTER        HASP-II VERSION 4.0
```